

AWARD NUMBER: W81XWH-14-1-0042

TITLE: See-What-I-Do: Increasing Mentor and Trainee Sense of Co-Presence in Trauma Surgeries with the STAR Platform

PRINCIPAL INVESTIGATOR: Juan P Wachs

RECIPIENT: Purdue University
West Lafayette, IN 47907

REPORT DATE: April 2016

TYPE OF REPORT: Annual

PREPARED FOR: U.S. Army Medical Research and Materiel Command
Fort Detrick, Maryland 21702-5012

DISTRIBUTION STATEMENT: Approved for public release; distribution is unlimited

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision unless so designated by other documentation.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE April 2016	2. REPORT TYPE Annual	3. DATES COVERED 3 Mar 2015 - 2 Mar 2016		
4. TITLE AND SUBTITLE See-What-I-Do: Increasing mentor and trainee sense of co-presence in trauma surgeries with the STAR platform		5a. CONTRACT NUMBER		
		5b. GRANT NUMBER W81XWH-14-1-0042		
		5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Juan P Wachs Email: jpwachs@purdue.edu		5d. PROJECT NUMBER		
		5e. TASK NUMBER		
		5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Purdue University West Lafayette, IN 47907		8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Medical Research and Materiel Command Fort Detrick, Maryland 21702-5012		10. SPONSOR/MONITOR'S ACRONYM(S)		
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for Public Release; Distribution Unlimited				
13. SUPPLEMENTARY NOTES				
14. ABSTRACT Our primary research objectives are to design, implement, and evaluate a working prototype that enables effective telementoring of a trainee surgeon by a remote mentor. This includes (1) a trainee-site subsystem for augmenting the view of the actual surgical field seamlessly by using a transparent display with illustrations of the current and next steps of the procedure, and (2) a mentor-side patient-size interaction platform with a gesture-based interface.				
15. SUBJECT TERMS- Nothing listed				
16. SECURITY CLASSIFICATION OF: a. REPORT U		17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 49	19a. NAME OF RESPONSIBLE PERSON USAMRMC
b. ABSTRACT U				c. THIS PAGE U

STANDARD FORM (SF) 298: A sample SF 298 is provided at <https://mrmc.amedd.army.mil/rrpindex.asp>. The abstract shall be provided in Block 14 and shall state the purpose, scope, and major findings and be an up-to-date report of the progress in terms of results and significance. Abstracts will be submitted to the Defense Technical Information Center (DTIC) and shall not contain proprietary information. Subject terms are keywords that may have been previously assigned to the proposal abstract or are keywords that may be significant to the research.

Pages shall be numbered. The number of pages shall include all pages that have printed data (including the front cover, SF 298, table of contents, and all appendices). Page numbers must match the numbering shown on the Table of Contents.

Purpose: Develop a framework that will enable increasing the mentor and trainee sense of co-presence through augmented visualization to facilitate surgical training and performance.

Scope: Optimal trauma treatment integrates different surgical skills not all available in military field hospitals. Telementoring can provide the missing expertise, but current systems require the trainee to shift focus frequently from the operating field to a nearby telestrator, they fail to illustrate the next surgical steps, and they give the mentor an incomplete picture of the ongoing surgery. We are addressing these gaps by developing STAR – System for Telementoring with Augmented Reality.

Major Findings: This year's main focus was on the mentor side interaction setting using a large tabletop display and different interaction methods, where medical experts (trauma surgeons), either working individually or as a team, are required to guide a novel general surgeon through a fasciotomy procedure. The experiment took place during a visit to Eskenazi Hospital in Indianapolis (USA), where 12 subjects were introduced to a telementoring situation where they would have to guide a surgeon with less experience through a four compartment leg fasciotomy. All actions executed by each subject were categorized between touch-based interaction, touchless interaction, or using a tool. By providing the ability to make touch-based annotations, use mid-air gestures and manipulate tools, this work attempts to enhance the sense of physical co-presence on the mentor side, conveying physical expressions to the trainee which are occurring remotely. Incorporating interaction methods such as gesture recognition and tool manipulation to basic telestrator capabilities such as drawing annotations, helps the mentors engage in the task while enhancing their sense of co-presence. The proportion of use among them (touch-based, mid-air gestures and tool placement) was found to be around 40-40-20. The use of large interaction tables in a collaborative telementoring setting allowed to observe team dynamics similar to the ones observed in an operating room like: anticipating a teammate's action and the handling and passing of tools. In addition, during this year, we validated our integrated simulated transparent display prototype. The prototype is a hand-held, self-contained system that acquires 3D geometry of the scene being viewed, tracks the user's head position in real time, and renders imagery of the scene from the user's viewpoint to achieve a transparent display effect.

TABLE OF CONTENTS

	<u>Page No.</u>
1. Introduction	
2. Keywords	
3. Accomplishments	
4. Impact	
5. Changes/Problems	
6. Products	
7. Participants & Other Collaborating Organizations	
8. Special Reporting Requirements	
9. Appendices	

1. INTRODUCTION:

Our primary research objectives are to design, implement, and evaluate a working prototype that enables effective telementoring of a trainee surgeon by a remote mentor. This includes (1) a trainee-site subsystem for augmenting the view of the actual surgical field seamlessly by using a transparent display with illustrations of the current and next steps of the procedure, and (2) a mentor-side patient-size interaction platform with a gesture-based interface.

2. KEYWORDS:

Augmented reality, telementoring, telemedicine, annotation anchoring, transparent display, surgical training, co-presence, simulation, tele-existence.

3. ACCOMPLISHMENTS:

The PI is reminded that the recipient organization is required to obtain prior written approval from the awarding agency Grants Officer whenever there are significant changes in the project or its direction.

What were the major goals of the project?

Specific Aim 1:

Implement transparent display (03-Mar-2014 – 03-Aug-2015) 100%
Achieve a visual overlay of info. from the mentor (03-Mar-2015 – 03-Mar-2016) 90%
Experimental Design 1: trainee subsystem (03-Apr-2016 – 03-Mar-2017) 40%

Specific Aim 2

Develop a gesture-based interaction system (03-Mar-2014 – 03-Aug-2015) 90%
Experimental Design 2: Gather gesture set (03-Apr-2015 – 03-Mar-2016) 100%
Experimental Design 3: Mentor subsystem (03-Oct-2016 – 03-Mar-2017) 35%

What was accomplished under these goals?

Major Activities: Research, develop, and assess a transparent-display augmented-reality system that allows the seamless enhancement of a trainee surgeon's natural view of the surgical field with annotations and illustrations of the current and next steps of the surgical procedure.

Specific Objectives

Task 1.1- Implement transparent display

Subtask 1.1.1: Evaluate tablet computer configuration

Improvement of current telementoring system and integration with new mentor system

As our team has worked on implementing a new version of the mentor system that incorporates a full-size interaction, we have taken the opportunity to improve the communication of data between the trainee and mentor systems. In previous experiments, the trainee and mentor systems were connected by a local network that allowed for high-bandwidth communication. However, in a real-world scenario -- and in the experiments we plan to conduct in the future -- the mentor and trainee systems will need to be truly remote and networked over the Internet. As a result, the question of bandwidth becomes more important.

In our telementoring system, video frames of the operating field at the trainee site are captured by the trainee's tablet system and transmitted to the remote mentor's system. Previously, these frames were sent as images in PNG format. As a result, the filesize of each frame was large: approximately 200 KB for even a low-resolution image of 640x400. These file sizes were impractical for streaming at real-time rates in a setting where the two systems were not located on the same local network.

To resolve this, we used the popular “ffmpeg” libraries for video encoding and decoding. On the trainee tablet system, we added additional functionality that would encode each frame using a video codec before delivering the encoded bytes over the network. On the mentor system, the incoming bytes were decoded using the same codec to yield the video frame. The choice of codec is important, as there is always a tradeoff between how much a frame can be compressed, the resulting quality of the encoded frame, and the computation time needed to encode each frame. Because the trainee tablet system must encode the acquired frames before transmitting them, the computation time is especially important to consider, and certain popular codecs such as MPEG4 were found to take too long to achieve real-time encoding.

We found that the MJPEG codec resulted in a good balance between computation time, frame filesize, and image quality. On average, each frame only requires 20 KB, which is superior to previous frame sizes of 200 KB. The trainee system is able to encode and transmit these frames to the mentor system at interactive rates (~15-20fps), which is sufficient for the mentor to oversee the operation as it proceeds. This is an important step that will allow us to continue with experiments to verify the validity of our telementoring approach.

Research into simulated transparent displays

Conceptual overview of simulated transparent displays

In this section we provide an overview and summary of our work into research simulated transparent displays. A simulated transparent display is a conventional opaque display that alters in real time the image it displays, such that from the perspective of the user, the display appears transparent. It achieves this by capturing the geometry (3D depth and color) of the part of the scene that is occluded by the display, and rendering the geometry from a known user viewpoint (either acquired in real time through eye tracking or by assuming a fixed user viewpoint).

Without a simulated transparent display, the mentee's view of the operating field appears distorted because the video displayed on the tablet screen is video taken from the camera's point of view. Because there is a difference between the camera's viewpoint and the mentee user's viewpoint, objects in the operating field area (such as hands, surgical instruments, and organs) may appear in a different location or with a different scale than what the mentee user would expect. This mismatch can impair the hand-eye coordination of a surgeon when conducting a surgical operation.

Truly transparent displays do exist, but their usefulness for surgical telementoring in austere environments is limited. First, current transparent displays remain partially opaque at all times, which would lead to a darker view of the operating field for a mentee user. Second, using a truly transparent display would require computation of the mentee system to be done either remotely or using a less compact form factor than the tablet devices we use. Such form factors would be less than ideal for the austere environment of a forward operating base.

For these reasons we have been continuing our research into simulated transparent displays. In the following sections we give an overview of our latest prototype display and its components, as well as an extended analysis of the quality of the transparent effect we achieve. This analysis, which consists of both theoretical error bounds from available sensors and empirical measurements of transparency error from real-world imagery, helps us evaluate which aspects of the transparent display should be improved next to yield the greatest benefit.

Implementation of simulated transparent display

In this section we provide an overview of the simulated transparent display prototype, described in earlier reports, used for our analysis. The prototype is a hand-held, self-contained system that acquires 3D geometry of the scene being viewed, tracks the user's head position in real time, and renders imagery of the scene from the user's viewpoint to achieve a transparent display effect.

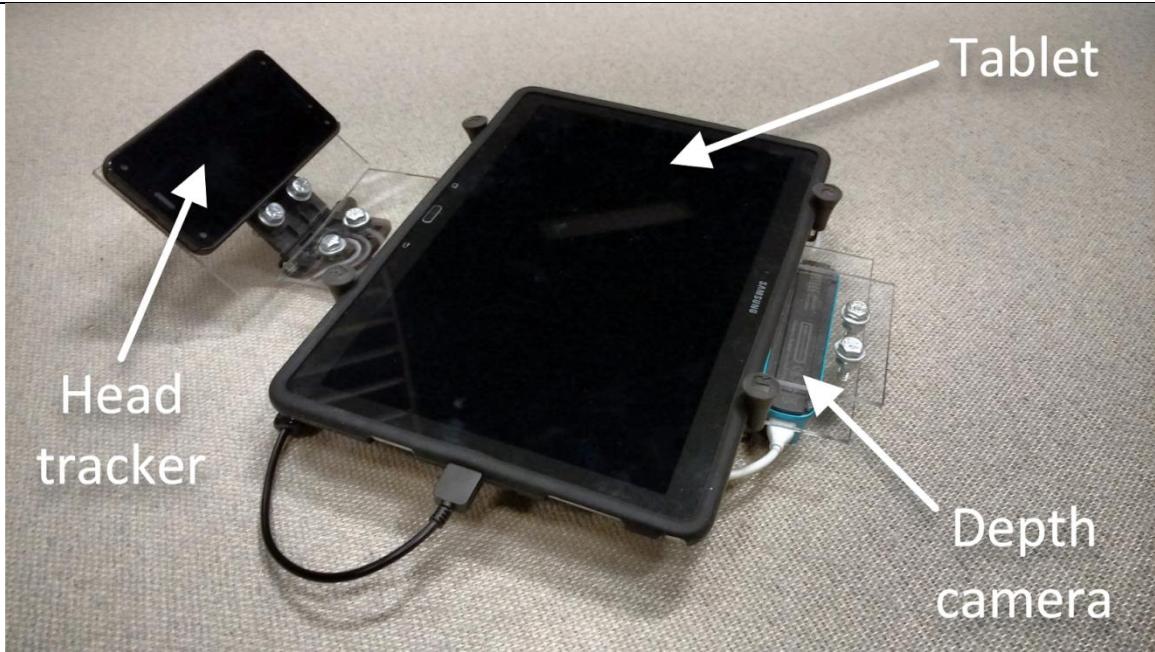


Figure 1: The components of the simulated transparent display used for our analysis.

Figure 1 shows the simulated transparent display prototype that we used for our analysis. The tablet display is the Samsung Galaxy Tab Pro 12.2-inch Android tablet that we have used for our STAR telementoring systems in the past. The depth camera is the Structure Sensor, which is an IR emitter/sensor that generates a depth map. The head tracker is the Amazon Fire Phone: an Android smartphone that uses its four front-facing cameras to triangulate the user's current head position with respect to the phone.

First, color is acquired from the tablet's color camera and depth data is acquired from the Structure Sensor. Color and depth is registered by finding a rigid transformation between the two cameras, such that for a particular location in the depth map, the corresponding color is known. Second, the head tracker finds the user's current head position and delivers it to the display. Finally, the geometry is rendered from the tracked head position. Figure 2 shows a first-person image showing the transparent display effect of our prototype.

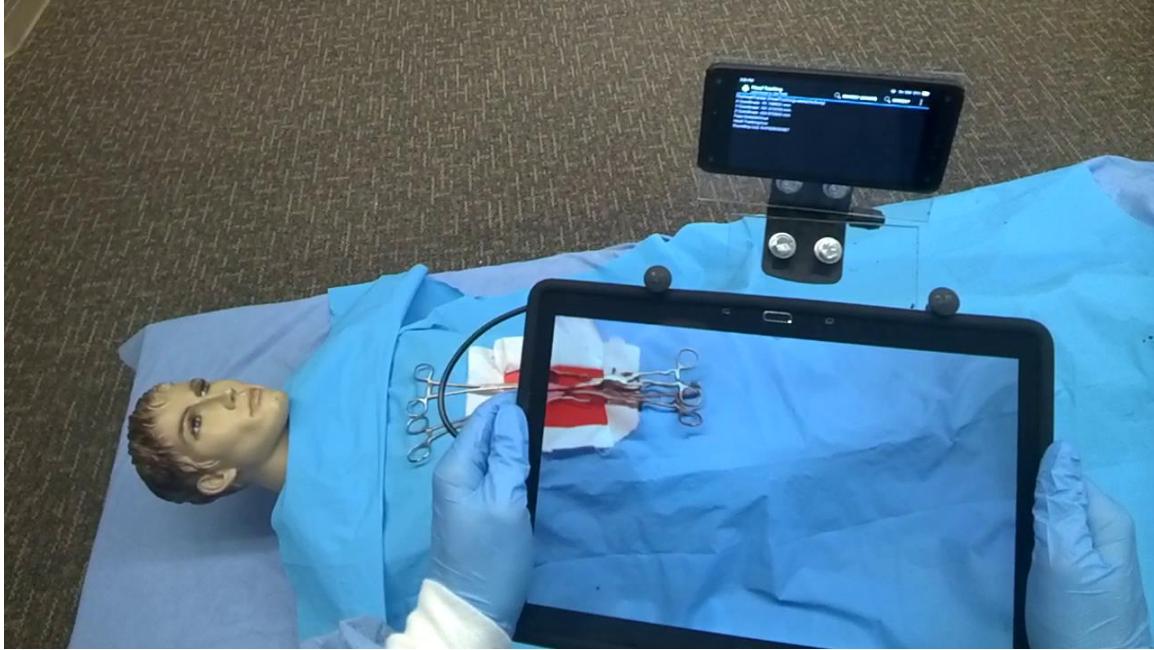


Figure 2: First-person image of transparent display effect.

Quantitative analysis of simulated transparent display quality

Before proceeding with additional incremental improvements to our simulated transparent display prototypes, we first defined a measurement of “transparency error” to quantify how closely the image rendered on a simulated transparent display resembles what the user would see if the device were not there. This is an important measurement because it allows us to not only evaluate how well our system is functioning, but it allows us to determine which kinds of technical improvements would lead to the greatest perceptual improvement for the user. As a result, we can more effectively motivate the next steps of our research. In this section we define a measurement of transparency error, we provide theoretical analysis of transparency error given various sensor error ranges, we show empirical transparency errors for real-world scenes, and we discuss how this analysis motivates the direction of future research.

Theoretical transparency error

We define the transparency error ϵ at a point p on the simulated transparent display as

$$\epsilon = |p - p_0| / d$$

The numerator is the distance in pixels between the actual position p and the correct position p_0 of the scene 3D point imaged at p , and d is the length of the diagonal of the display in pixels. If the transparency error ϵ is 0, then the transparency effect is perfect because the user perceives no change in the position of scene objects when the display is present or not present.

First, we provide a theoretical analysis of the effect of depth acquisition quality on the transparency error. Our transparent display’s depth sensor use active, structured-light depth acquisition, which is not always accurate. Moreover, missing depth data is interpolated with

nearby depth data, which is only an approximation. Figure 3 shows the maximum transparency error for our transparent display prototype as a function of depth acquisition error. The scene is assumed to be 1m away behind the display, and the user viewpoint is assumed to be 0.5m away in front of the display, which is a typical use case. Typical real-world depth acquisition errors are in the 10mm range, which corresponds to a low transparency error of 0.3%.

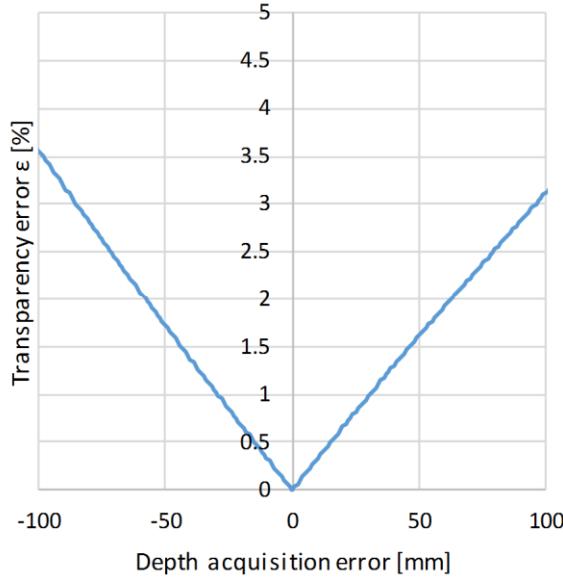


Figure 3: Transparency error as a function of depth detection error. A negative / positive depth detection error means that the scene is farther / closer than the acquired depth indicates.

Another cause of transparency error is inaccurate head position tracking of the user. If the user's true viewpoint is different from the device's perceived user viewpoint, then the image displayed on the device's screen is no longer accurate. There are two main kinds of tracking error: error in x/y and error in z. Error in x/y means that the predicted user viewpoint is shifted left, right, up, or down in relation to the tablet display. Error in z means that the predicted user viewpoint is shifted closer to or further from the tablet display.

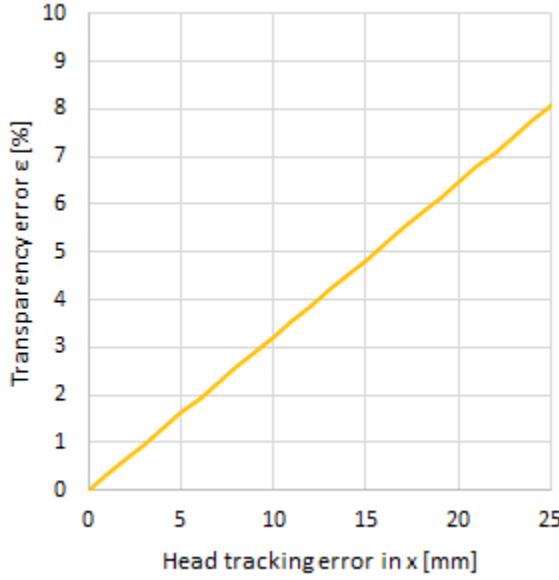


Figure 4: Maximum transparency error as a function of user head tracking error in x (similar for y).

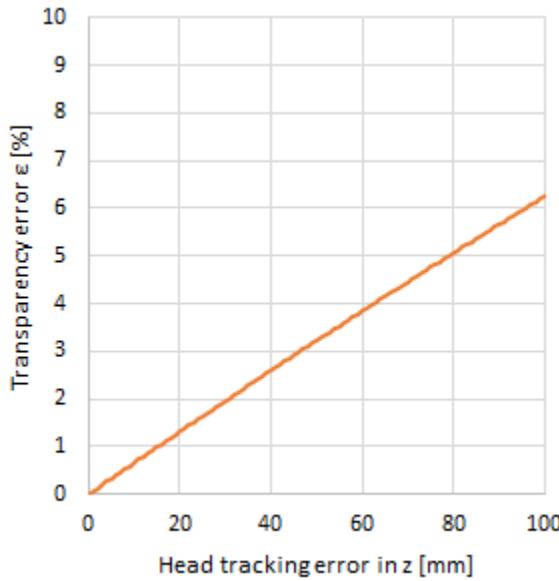


Figure 5: Maximum transparency error as a function of user head tracking error in z.

Figure 4 shows the maximum transparency error as a function of user head position tracking error in x (similar for y); Figure 5 shows the maximum transparency error as a function of head tracking error in z. Negative head tracking errors in z indicate that the true head position is farther from the display than tracked, while positive errors indicate that the true head position is closer to the display than tracked. The user's head is assumed to be 0.5m away from the displays; the scene is assumed to be 1m away, which is a typical use case. The transparency error depends more on the x than the z head tracking error. Head tracking is

typically accurate to less than 10mm in x and 30mm in z, which translates to maximum transparency errors of 3.2%.

Empirical Measurements

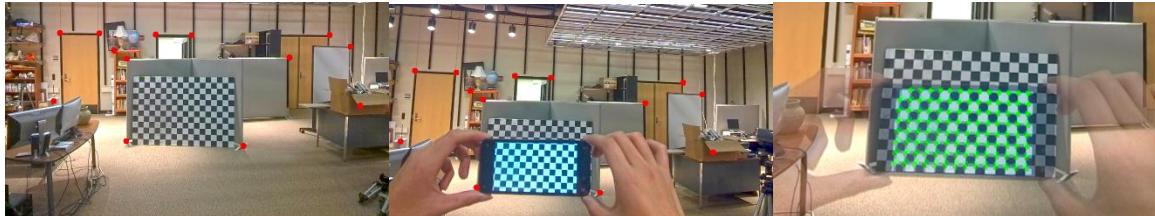


Figure 6: Empirical transparency error measurement. Left: Reference image of the scene taken by Google Glass. Middle: Image taken by Google Glass while using the transparent display.

The red dots illustrate manually selected salient features in the region outside of the transparent display, which are used to align the two images. Right: Overlay image where the actual transparency error is measured using manually selected correspondences (green dots) in the region covered by the transparent display.

We compared these theoretical error bounds against imagery of real-world scenes as seen through the simulated transparent display. Images were taken by having the user wear the Google Glass head mounted camera. First, the user acquires an image I_1 of the scene using the Google Glass camera (Figure 6, left). Next, the user acquires a second image I_2 of the scene while holding up the simulated transparent display, which has been calibrated to generate a transparent effect for the viewpoint of the Google Glass camera (Figure 6, middle). Since the user is likely to tilt their head slightly as they acquire the two images, I_1 and I_2 have to be first aligned using the region outside the transparent display. We align the two images by computing a homography between I_1 and I_2 using manually selected corresponding salient features in the region outside the display. The homography is used to compute an overlaid image I_3 (Figure 6, right). The transparency error is then computed by measuring the distance between manually selected corresponding features in I_3 that are within the transparent display region.

Our theoretical analysis predicts a transparency error of about 5% with our current sensors. Table 1 gives actual transparency error values for our most recent prototype. These empirical results show that our prototypes achieve a good transparency effect. The small error values (1.6%, 3.1%) indicate that the actual head tracking errors are smaller than the upper bounds used in the theoretical analysis above.

Image I₁		
Image I₂		
Error ϵ	3.1%	1.6%

Table 1: Empirical measurements for our simulated transparent display.

Discussion and conclusions from analysis

The empirical results described above suggest that the transparency quality is on the order of what we can expect, given the sensors we are using. Our theoretical analysis provides direction as to which parts of the simulated transparent display can be enhanced to achieve the greatest improvement in transparency quality. Regarding depth acquisition, our current sensors are of sufficient quality as long as they acquire some depth value. That is, for some point where a depth value is sampled, that depth value is sufficiently accurate. However, in areas of disocclusion (where operating field elements are not visible to the depth sensor, but should be visible from the user’s viewpoint), the missing depth values lead to increased transparency error. Methods of merging together previous depth frames into a coherent map of the operating field will be needed to overcome this.

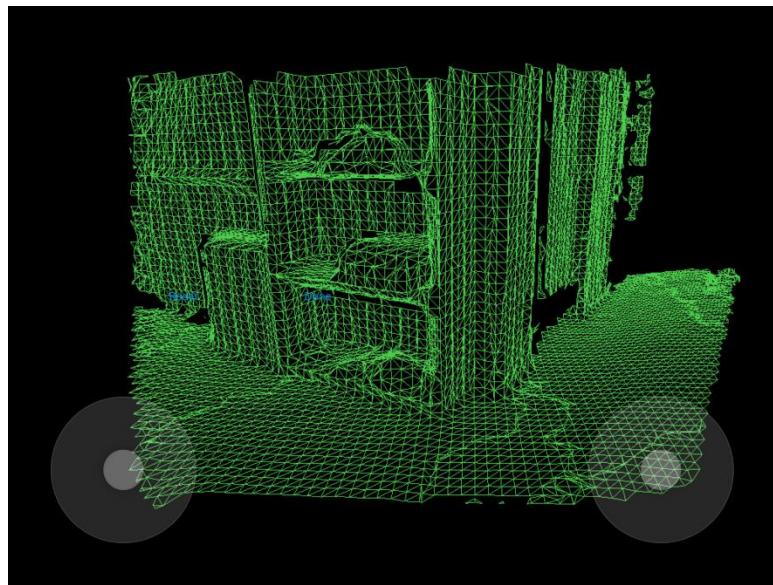
Regarding head tracking, the current head tracking approach is reasonably accurate, though it can be incrementally improved with more precise camera-based tracking systems that detect eye position rather than a generic “head position.” The main limitation, though, is that the simulated transparent display’s rendered image is only correct for a single viewpoint (i.e. only one of the user’s eyes). When the user views the display with the other eye, this is analogous to viewing the display with a high amount of head tracking error in x (as described in Figure YYY). Possible solutions to this involve the use of autostereoscopic displays that can display different images to each eye without the use of eyewear. We have investigated current autostereoscopic approaches; however, this is still an emerging research field with no clear-cut consumer solutions. Because research into display technology is less within the scope of our work, we are for the time being setting aside this question, so we can focus our efforts on areas more directly related to the field of surgical telementoring.

Explorations into combining 3D scanning with transparent displays

The aforementioned simulated transparent display prototype only renders scene geometry that is currently within the field of view of its sensors. Only the most recent color and depth data is used. As a result, large disocclusions result when the user views the scene through the display at an oblique angle, because scene geometry for such areas is not saved. To resolve this, we are investigating the use of SLAM (Simultaneous Localization and Mapping) approaches that can merge multiple depth and color maps into a single 3D mesh.

We have implemented an exploratory prototype that uses the Structure Sensor on an iPad to create a fixed-viewpoint simulated transparent display that progressively builds a 3D model of the scene. The purpose of this particular prototype is not to be a fully-functioning simulated transparent display; instead, it is a testbed to evaluate the feasibility of 3D scanning and SLAM approaches for future integration into the telementoring system.

The Structure Sensor that we use for depth acquisition was originally designed to work with iOS devices, and its manufacturers provide a set of 3D mapping libraries on iOS. When a Structure Sensor is attached to an iPad, these libraries allow an app to capture multiple keyframes of color and depth data and combine them into a single textured mesh. Real-time mesh generation is supported.



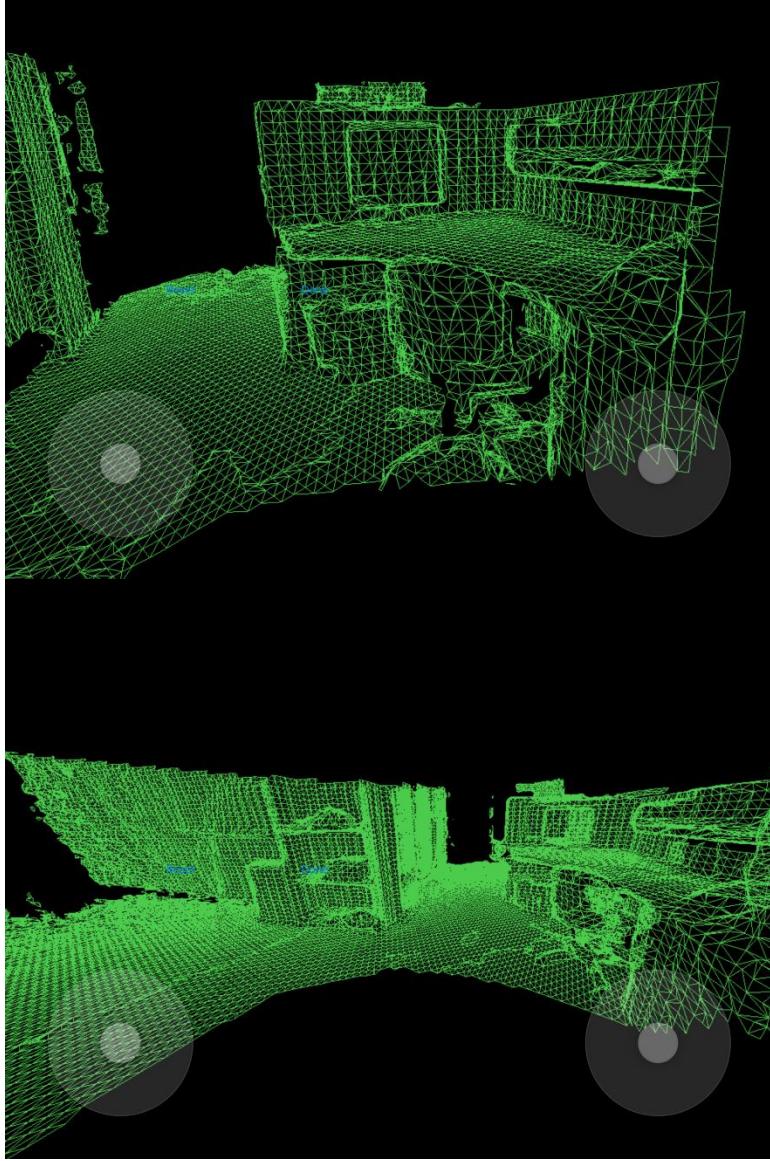


Figure 7: Screenshots of the test iOS 3D scanning prototype. Circular icons in the lower corners are virtual joysticks to manipulate the user viewpoint. Top: A 3D mesh of an office scene, initially acquired from a single viewpoint. Middle: The 3D mesh is extended as the tablet moves. Bottom: A panoramic 3D mesh acquired by the prototype.

We created an iOS app for an iPad Air 2 tablet that builds a mesh as the tablet moves with relation to the scene. Figure 7 shows screenshots from the app. We added a touch-based user interface that allows a virtual user viewpoint to be defined by two virtual joysticks. The left joystick controls the virtual user viewpoint in the X and Y dimensions (parallel to the tablet screen), and the right joystick controls the virtual user viewpoint in the Z dimension (normal to the tablet screen).

The acquired 3D geometry is rendered as a wireframe mesh using the same transparent display rendering techniques we use in our previous prototypes. To perceive the transparent display effect, the user first holds the tablet in a fixed position with respect to their head and manually

adjusts the virtual viewpoint until the position and scale of the rendered geometry aligns with the user's real-world view of the surrounding scene. Then, as the user moves the tablet while remaining in the same relative viewpoint, the image on screen continues to appear aligned.

The tablet is able to render the geometry at real time rates, and the 3D mesh is constructed in real time without needing to do offline pre-processing. This indicates that such SLAM approaches will be feasible when being used in the context of surgical telementoring, where 3D geometry of the patient should be captured and available to the system in real time. One limitation of this particular prototype is that it lacks the viewpoint-tracking ability of previous prototypes; the user must manually adjust the viewpoint using the user interface. However, head tracking could be integrated in the same way as previously implemented: by attaching a Fire Phone and transmitting head tracking data over Bluetooth.

Specific Objectives

Task 1.2 - Achieve visual overlay of information

Subtask 1.2.2: Generate illustration of next steps of surgery through simulation

One goal of our project is to not only provide the mentee with a visualization of the current step that should be performed, but also to simulate imagery of the operation beyond the present moment. Our work here is divided into two sections. The first section describes our work on illustrating past steps in the operation to the mentee. The second section describes our planned and initial steps toward simulating future steps for the mentee to visualize.

Showing imagery of prior steps

Showing earlier stages of the current operation is potentially useful to the mentor and mentee. For example, the mentee may wish to gain additional context for the mentor's current instructions by comparing them with previous instructions. Also, having a record of the mentor's instructions would help with post-operative debriefs by allowing both surgeons to refer to the provided instructions when evaluating what went well during the surgery or what could be improved. At the same time, merely recording a video of the surgery may be cumbersome for a mentee to use as a reference during the operation itself. Rather, a semantic-based approach that saves previous instructions when there was a significant change in the annotations allows the mentee to more quickly traverse past imagery.

We modified our trainee tablet system so that, whenever a new annotation from the mentor was received, a screenshot of the operating field was captured. This screenshot includes both the background image captured by the tablet and the annotations as rendered on the screen. The screenshots are available in the user interface, allowing the user to navigate forward and backward between each of the screenshots while the real-time imagery from the operating field continues in the main window.

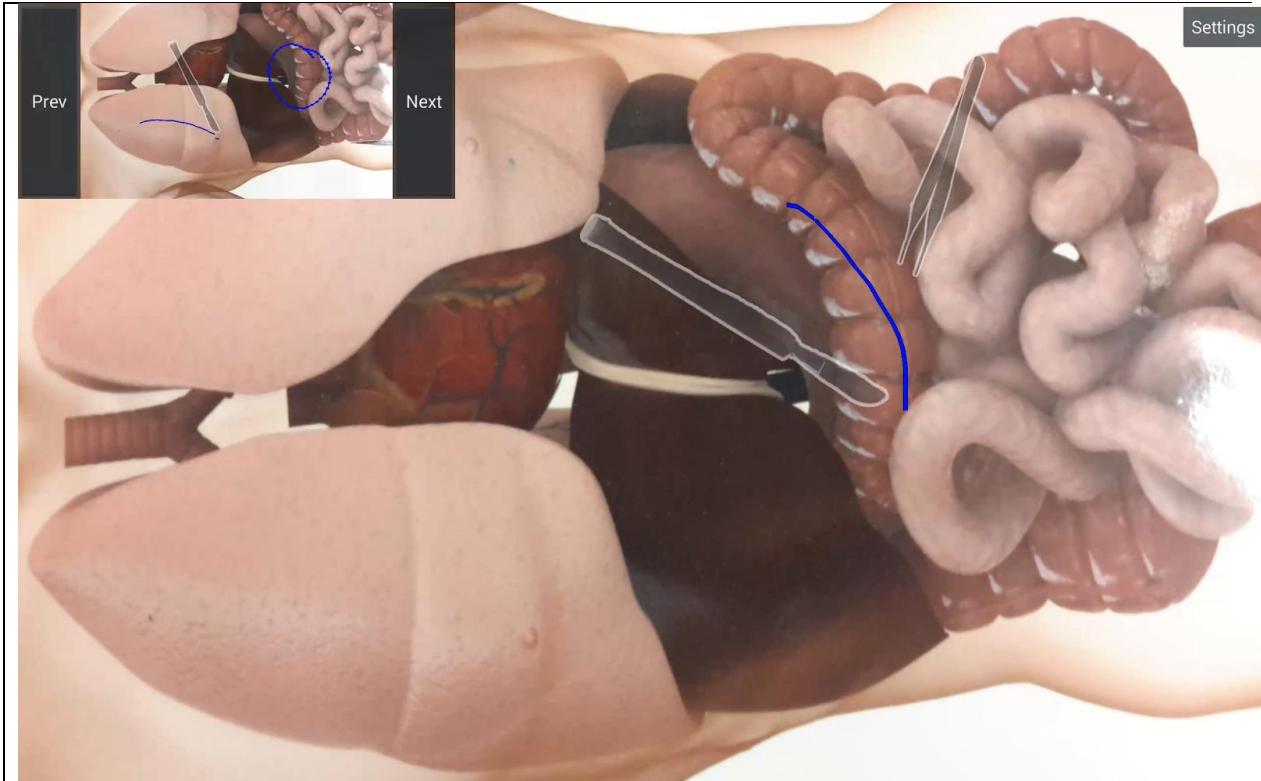


Figure 8: Example visualization of previous steps on the trainee tablet's user interface. The current annotations and operating field are visible on the main window, while older screenshots are visible in the upper left.

Figure 8 shows an example image of the visualization of previous steps. For the purposes of testing and development, the user interface uses on-screen buttons to navigate through the captured screenshots. In a surgical setting, it would of course be infeasible for the mentee to touch the screen, but such a system could be enhanced with voice controls for hands-free interaction.

Visualization of future steps in surgery

Visualization of future steps in the surgery can benefit the mentee by providing additional context for current instructions. If the mentee is able to see what the expected result is of an action, the mentee will be able to perform the action more accurately. These visualizations of future actions must be overlaid directly onto the relevant areas of the operating field, to prevent issues with focus shifting.

In this section we describe two ongoing approaches to future visualization that we are investigating. First, we have implemented an animated incision annotation that runs on the mentee tablet system. Second, we propose an approach to using video imagery of prior surgical operations as parameterized overlays for future visualization.

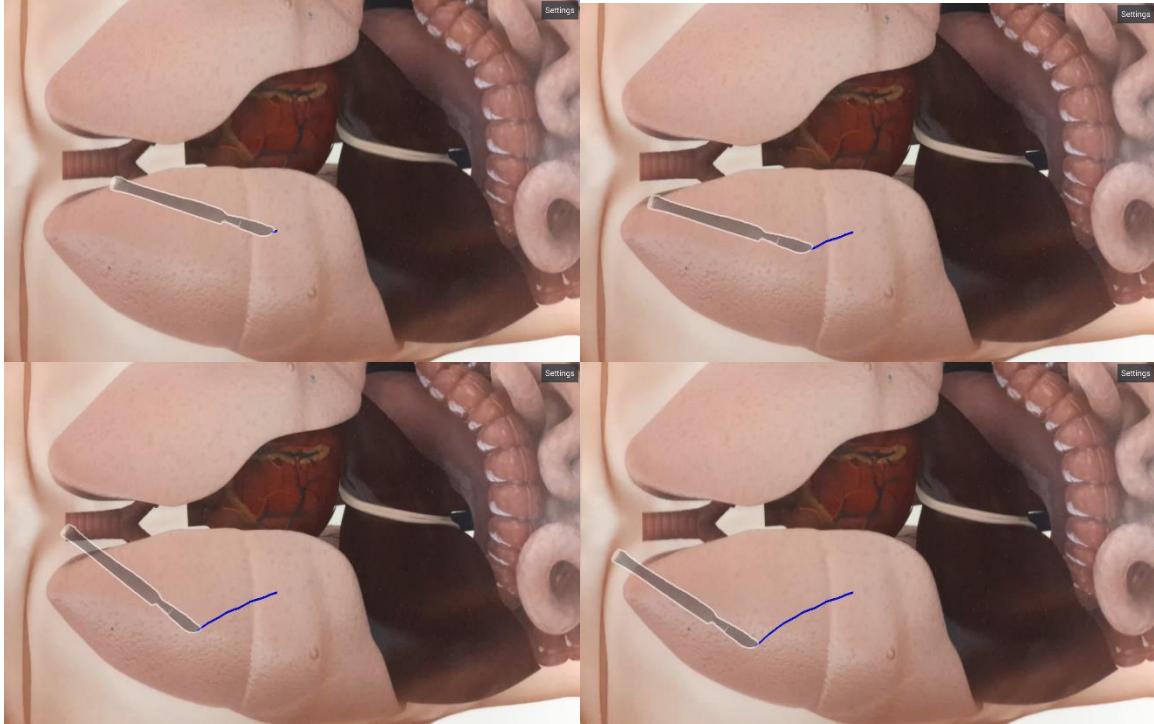


Figure 9: Example of animated incision annotation, as seen by the mentee system. Each image displays the same annotation at different timestamps.

We have adapted our existing support for polyline annotations to create an “animated incision annotation.” The mentor user first creates an animated incision annotation in the same way as creating a polyline annotation: i.e. by drawing a line using the mentor system’s touch-based user interface. When this annotation is transmitted to the mentee system, it displays not as a static line but as a line that progressively extends along the mentor-defined path in a looping animation. In addition, a scalpel sprite is automatically drawn along the endpoint of the path, such that the scalpel appears to be making the incision. Figure 9 shows an example of this kind of animated annotation.

While an animated annotation can help a mentee visualize how to complete a particular telementored task, we also want to create visualizations that show a future state of the surgery. Here we define a framework that we will implement to achieve this goal. First, prior video imagery of relevant stages of a surgical operation are compiled into a database of video clips. For example, each incision of a reference fasciotomy is recorded and saved individually. Second, anchor points on each step are defined; for example, the video clip of an incision would have anchor points defined at the pixel locations where the incision begins and ends. Third, these video clips are instantiated as animated annotations in the STAR system, with the mentor defining the corresponding anchor points in the operating field for the current surgery. The result of this approach is that existing video references can be overlaid directly onto relevant areas of the operating field, scaled/rotated/repositioned into the correct orientation.

Major activities: Research, develop and assess a patient-size interaction platform where the mentor can mark, annotate, and zoom in on anatomic regions over a projected image or on a multipoint-touch screen.

Task 2.1- Develop a gesture-based interaction system

Subtask 2.1.1 – One-Shot Learning Gesture Recognition

Gesture Analysis on Data gathered in Eskenazi Hospital Visit

During our visit to Eskenazi Hospital, subjects were recruited to act as mentors in a telementoring scenario where they needed to guide and assist a surgical trainee performing a four compartment fasciotomy of the leg in a remote location. The setting is shown in Figure 1. Each mentor was asked to stand in front of the display and give instructions based on the images received from the trainee's site (which at the moment of the experiment is fictional, and the images are previously gathered).

Throughout the procedure, they had three modes of interaction available: they could perform annotations by drawing on the screen, they could perform air gestures (which were detected in a "Wizard of Oz" methodology and a member from the research team executed the desired instruction), or they could place physical tools on top of the screen. As the mentors performed the procedure, both a Kinect and a video camera were recording their movements for posterior analysis. After the experiments were conducted, the video recordings were analyzed for each subject.

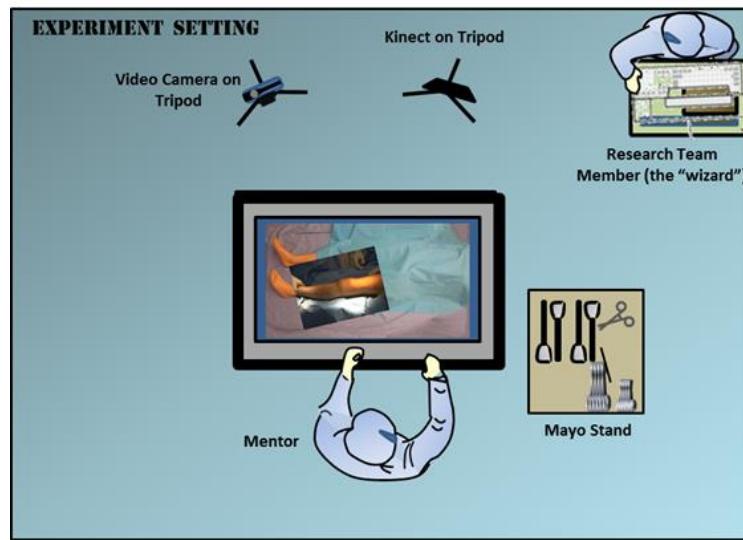


Figure 1. Setting for Experimental Design 2: Gather Gesture Dataset

All actions executed were categorized between touch-based interaction, touchless interaction, or using a tool. Total number of actions were computed for the entire procedure, as well as how many times each participant used each mode of interaction; consequently, total values were obtained of how many of their actions were expressed using each interaction method.

Additionally, each gesture performed by each participant to elicit a specific command, regardless of the interaction mode, was identified and accumulated, to measure intuitiveness,

popularity, and agreement among the participants. Equations for the intuitive index a_{ij} and popularity q_i are shown below. The indices i, j, k represent gesture, command and subject respectively. The entry a_{ij} represents the number of participants selecting gesture i to execute command j . Values for q_i represent the number of participants that selected gesture i , giving a measure of popularity.

$$a_{ij} = \sum_{\substack{k=1, \dots, K \\ \text{participants}}} a_{ij}^k \quad q_i = \sum_{\substack{j=1, \dots, C \\ \text{commands}}} a_{ij}$$

Agreement was measured using a ratio between actual agreements over all possible agreements for each gesture selected. The equation is shown below. The mean rate among all gestures gives an indication of the measure of agreement of the group studied.

$$S_i = \frac{\sum_{j=1, \dots, C} a_{ij} (a_{ij} - 1)}{(\sum_{j=1, \dots, C} a_{ij})(\sum_{j=1, \dots, C} a_{ij} - 1)}$$

For 9 subjects, the data collected shows 24 commands were used, and 117 gestural responses were made, 49 of them being unique. Some of these results are displayed on Table 1. Values for popularity q are shown both in the table and Figure 2.

Table 1. Aggregate intuitive indices

Type	Gesture / Action /	Anatomical Marking	Area of Interest	Cut	Distance Measurement	Erase	Incision	Muscle emerging	Palpate	Release	Rotate image	Separate	Spread Incision	q	S_i
Touch	Draw line	2	1	0	0	0	8	0	0	0	0	0	0	12	0.44
Tool	Motion with scissors	0	0	5	0	0	0	0	0	1	0	1	0	7	0.48
Touch	Tap Screen	3	1	0	0	0	0	0	3	0	0	0	0	7	0.29
Touchless	Pointing pose and line	0	2	0	0	0	4	0	0	0	0	1	0	7	0.33
Touchless	Pointing pose	1	4	0	0	0	0	0	0	0	0	0	0	5	0.60
Touchless	C-shape hand moving	0	1	0	0	0	0	0	2	0	0	0	0	4	0.17
Touchless	Index in hook pose and moving	0	0	0	0	0	0	0	2	0	0	2	0	4	0.33
Touchless	Open hand hover	0	0	0	0	0	0	0	1	1	0	1	0	4	0.00
Tool	Point with scissors	0	2	0	0	0	0	0	0	1	0	0	0	3	0.33
Touch	Draw wavy line	1	2	0	0	0	0	0	0	0	0	0	0	3	0.33
Touchless	Swipe hand palm facing screen left to right	0	0	0	0	2	0	0	0	0	0	0	0	3	0.33
Touchless	Pointing with two fingers	0	0	0	0	0	1	0	1	0	0	0	0	3	0.00
Touchless	Line motion	1	1	0	0	0	0	0	0	0	0	0	0	3	0.00
Touchless	From open hands together to spread apart	0	0	0	0	0	0	0	0	1	0	0	1	3	0.00
Touchless	reverse pinching motion	0	0	0	0	0	0	0	0	0	0	0	3	1.00	
Tool	Place retractor on table	0	0	0	0	0	0	0	0	0	0	0	2	2	1.00
Tool	Motion with scalpel	0	0	0	0	0	1	0	0	0	0	0	0	2	0.00
Tool	Motion with retractor	0	0	0	0	0	0	0	0	0	0	0	2	2	1.00
Touch	Swipe finger	0	1	0	0	0	1	0	0	0	0	0	0	2	0.00
Touch	Two-finger rotate	0	0	0	0	0	0	0	0	0	2	0	0	2	1.00
Touchless	Both hands press down	0	0	0	0	0	0	0	1	0	0	0	0	2	0.00
Touchless	Pointing pose and circle	0	0	0	0	1	0	0	0	0	0	0	0	2	0.00
Touchless	Pose thumb and index apart	0	0	0	2	0	0	0	0	0	0	0	0	2	1.00
Touchless	One hand press down	0	0	0	0	0	0	0	0	0	0	0	0	2	0.00
Touchless	Pose pressing with thumb	0	0	0	2	0	0	0	0	0	0	0	0	2	1.00

The most popular gesture performed was drawing a line with $q = 12$, which 89% of the participants associated to the command make an incision. Out of the 117 total gesture responses collected, the 49 considered unique represent 42% of all possibilities. By frequency examination, it was determined that 81% of the participants chose 55% of all gesture types.

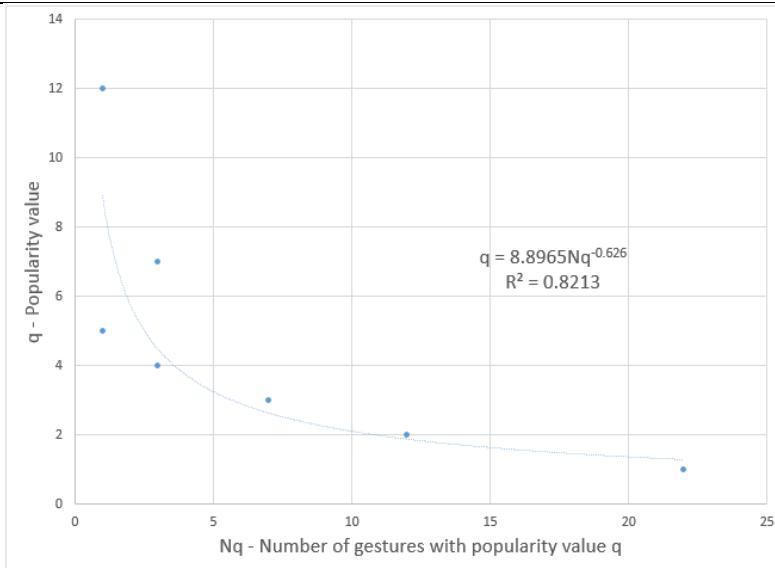


Figure 2. Gesture popularity graph

Regarding agreement, partial proportion S_i are shown in Table 1 for the most popular gestures. To illustrate the metric calculation, consider the “pointing pose” touchless gesture. This gesture was associated with commands “Anatomical Marking” by one participant and “Area of interest” by other four participants; these constitute 12 separate agreements, each participant agreeing with 3 others out of 20 possible agreement coming from all 5 participants using the gesture agreeing with each other. The overall agreement for the studied group was found at 36%.

$$S_5 = \frac{4 * 3}{5 * 4} = 0.6$$

Out of the 49 unique gestures performed for the fasciotomy experiment, there were 22 gestures only selected by one participant, resulting in 45% of the basic vocabulary a customized gesture for a given participant. This result further demonstrates the need for a system that can be personalized, providing a natural means of interaction for the mentor to provide instructions.

Gist of Gesture framework implemented using different classifiers

Using a vocabulary of 11 gestures, which include some actions observed during the Experimental Design 2, the “gist of the gesture” methodology was used to generate artificial observations of an example of each gesture to train three different classifiers. This was done to test the method itself regardless of the classification algorithm used at a given time.

Most of the gestures selected represent actions to manipulate the display system; others gestures relate to the manipulation of tools, such as pick, drop or cut. Among the used gestures are:

- **Zoom in:** the users start the gestures with their hands coming together around the center of their torso and moving away from each other towards the outside of the body.
- **Zoom out:** it is the opposite of zoom in, both in meaning and gestural action, in which the hands begin separated and towards the outside of the body and come together in the center of the torso area.

- **Rotate clockwise/counter-clockwise:** since these gestures are an opposing pair, the main difference relies on the direction of motion. Using the right arm bent at the elbow, the hand rotates toward the center of the body (for counter-clockwise) or towards the outside (clockwise).
- **Pick/Drop:** users elevate their hands to signal picking up a tool and lower their hands towards the outside of the body starting from an elevated position to signal dropping it.
- **Erase:** given a situation where an annotation was manually performed on the display, erasing gesture includes a circular motion for the hand in a plane facing the sensor.
- **Paste:** moving the right hand in a descending motion to reach the left hand to signal attaching a new image or annotation to the displayed on screen.
- **Previous/Next:** these gestures were included as some of the participants wished to advance on the experiments as the surgical instructions were already given and were expecting the next frame indicating the new circumstances where the “trainee surgeon” required assistance.
- **Cut:** some of the participants performed the air gesture to cut using their hand moving down towards the screen either to incise using a scalpel or scissors.

The input information is based on skeleton models tracked using Microsoft Kinect to detect a user and the movements made with their upper limbs. Based on one example from a gesture, the goal is to have a genuine representation such that an arbitrary observer could not be able to tell which was performed by a human and which was generated artificially. In order to achieve this goal, it is necessary to leverage on bio-mechanical features reflecting the physical and dynamic limitations of humans during gesture production.

Gestures are defined as a concatenation of movement phases, with the following distinctions:

- Movement phases are separated by abrupt changes in orientation, and changes in speed.
- Phase segmentation is invariant with respect to the duration of movement.
- Gestures are bound by a sequential order but time invariant.
- Gestures performed by humans show spatial generalization and orientation specificity.

Given such considerations, the selected features are: the number and location of inflexion points for each hand’s trajectory (an array of three-dimensional points), the type of curvature present between a pair of inflexion points (e.g. convex, straight, and concave), and the sequence of the movement described by the quadrant where each inflexion point is located with respect to the gesturer’s shoulder, considering the y-z plane as above and below the shoulder, and closer to or further away from the body centroid (this is an anthropometric feature). Adding meaningful variability to these features, allows for an expansion of the training data set while preserving the fundamental structure of the original gesture.

The one-shot learning approach based on “the gist of the gesture” can be summarized in the following pseudo-code:

Algorithm 1. Generate artificial observations from one sample

Input: 3D hand trajectory \mathbb{X} , 3D position of shoulder $\mathbf{x}_s = (x_s, y_s, z_s)$, number of artificial trajectories to generate N

Output: Set of artificial trajectories $\mathbf{a} = \{a_1, a_2, \dots, a_N\}$

1. Extract “gist of gesture”

1.1. Find M inflexion points \mathbf{x}_i in hand’s \mathbb{X} trajectory

$$\left. \frac{d\mathbf{x}^2}{d^2t} \right|_{x=x_i} = 0, i = 1, \dots, M$$

1.2. Determine convexity c_j for interval I_j between inflexion points \mathbf{x}_i and \mathbf{x}_{i+1}

$$I_j = \{x \mid x \in (x_i, x_{i+1})\}, j = 1, \dots, M-1$$

for $j = 1, \dots, M-1$ do:

$$c_j = \text{sign}(\ddot{f}(x)), x \in I_j$$

1.3. Determine quadrant location $q(\mathbf{x}_i)$ to each inflexion point \mathbf{x}_i based on shoulder’s location \mathbf{x}_s

$$q(\mathbf{x}_i) = \begin{cases} I & y_i > y_s, z_i > z_s \\ II & y_i < y_s, z_i > z_s \\ III & y_i < y_s, z_i < z_s \\ IV & y_i > y_s, z_i < z_s \end{cases}$$

2. Generate N artificial observations $\mathbf{a} = \{a_1, a_2, \dots, a_N\}$

2.1. Compute variance estimation σ_k based on the original trajectory points p_i on each quadrant $q(\mathbf{x}_i)$

$$\sigma_k = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} (p_i - \mu_k)^2, p_i \in |q(\mathbf{x}_i)|_k \in \mathbb{R}^3 \text{ & } k = 1, 2, 3, 4$$

2.2. Generate GMM, denoted as Γ_i // add variability to each \mathbf{x}_i

$$\Gamma_i = \Sigma \sim N(\mathbf{x}_i, \sigma_k), i = 1, \dots, M, k = 1, \dots, 4$$

2.3. Sample each Γ_i to obtain a set of M inflexion points \mathbf{x}_i^*

$$\mathbf{x}_i^* \in \Gamma_i, i = 1, \dots, M$$

2.4. Smoothly connect \mathbf{x}_i^* and \mathbf{x}_{i+1}^* using c_j

$$a_l = \cup \text{arc}(\mathbf{x}_i^*, \mathbf{x}_{i+1}^*, c_j), l = 1, \dots, N$$

Artificial trajectories $\mathbf{a} = \{a_1, a_2, \dots, a_N\}$

The selected classifiers were Hidden Markov Models (HMM), Support Vector Machines (SVM) and Conditional Random Fields (CRF) given their popularity in the literature for gesture recognition applications. A one-vs-all scheme was maintained for all classifiers. Once the training set of artificial trajectories was generated, a feature representation based on the modulus and the angles representing the incremental changes in positions for each hand’s trajectory was used, resulting in a 6-dimensional vector. As discrete HMMs were used, an additional quantization step was necessary to reduce the dimensionality of the feature representation to a code. For the SVM, each was trained using the RBF kernel function. In the case of CRF, the training examples were encoded using BIO, to determine the beginning (B), inside (I), and outside (O) of a gesture.

The training dataset includes 300 artificially generated examples. The testing dataset is comprised by 30 examples gathered from 5 different users performing the gesture vocabulary six times each. Thus, one-shot learning is accomplished by training the classifiers on artificially generated instances. Said procedure enables the recognition of future instances of each gesture in the vocabulary.

In order to obtain the ROC interaction curves for each classifier, a free parameter was selected in each to vary and obtain different values for hit rate and false alarm. In the case of HMM and CRF, given that their configuration is intrinsically related to probabilities, the

parameter was assigned as the ratio between the highest and the second highest probability obtained from each classifier. In the case of the SVM, the selected parameter was the scaling factor in the Gaussian radial basis function kernel. These parameters were varied and used as threshold with three different values and the curves were completed with the two extremes: (0,0) and (1,1). The same parameter was used 3 times, dividing the dataset and reshuffling in groups of 10. Figure 3 shows the means in the obtained ROC curves for the three classifiers.

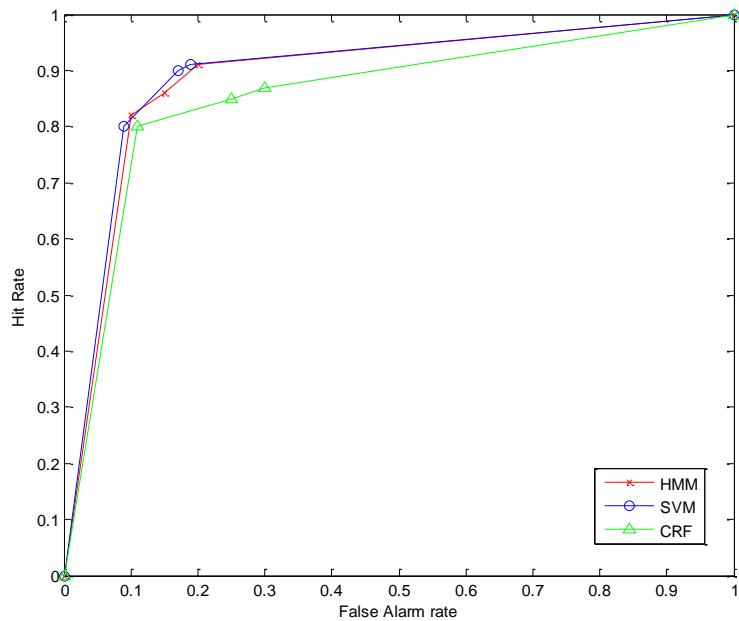


Figure 3. ROC curves for the three trained classifiers

Based on the obtained ROC curves, the area under the curve was calculated for all the classifiers resulting in a recognition of 97.05% for HMM, 97.2% for SVM, and 95.9% for CRF. These results show rather similar recognition performance for SVM and HMM while CRF shows slightly lower performance. The obtained results show the feasibility of the method, and the general accuracy to be the same regardless of the selected classifier.

Subtask 2.1.2 – Design a projection surface and interaction methodology

PQLabs Touch Overlay Installation and Usage

The requested PQLabs touch overlay arrived and its installation was done. The 65 inches frame overlay was installed over a screen previously purchased that was part of the lab equipment. This installation was done under a Windows 7 platform, although the drivers of the overlay are also available for MacOS and Linux (Ubuntu and Fedora). The Software Development Kit and examples of the overlay was downloaded from the company's webpage. The same demos used in the previous report were replicated using this overlay, proving that the calibration of the equipment was well done and that the experience of using the system was the same that was obtained when the demos were previously performed. An image of the patient-size multipoint-touch screen (Figure 4) is provided below.



Figure 4. 65" Screen with a PQLabs Touch Overlay System

Once the overlay was installed, a touch controller module was programmed using the SDK and following the examples provided by PQLabs. With this module, the touch overlay could not only detect when a touch event was performed on the screen, but the type of the event performed. Once the event was correctly detected, the system would inform the other modules about the event so that the right processes could be started after the event was done.

Because the system allowed the detection and interpretation of a variety of touch events (either single or multipoint), a touch gesture dictionary was created. Using these events, the system can perform all the tasks that are required at the moment, like drawing, editing and selecting lines and tool annotations and controlling the graphic user interface of the system. A summarized version of the created dictionary is presented in the following table:

Table 2. Touch gesture dictionary

Touch event name	Event explanation
TG_TOUCH_START	The overlay recognized that a touch event initiated.
TG_DOWN	A single finger touched and remained on the screen.
TG_CLICK	A single finger click was done on the screen.
TG_MOVE_RIGHT	A single finger is moving to the right on the screen.
TG_MOVE_LEFT	A single finger is moving to the left on the screen.
TG_MOVE_DOWN	A single finger is moving downwards on the screen.
TG_MOVE_UP	A single finger is moving upwards on the screen.
TG_TOUCH_END	All the fingers are no longer on the screen.
TG_ROTATE_CLOCK	One finger is rotating clockwise around another one.
TG_ROTATE_ANTICLOCK	One finger is rotating anticlockwise around another one.
TG_SPLIT_APART	Two fingers are moving away from each other.

TG_SPLIT_CLOSE	Two fingers are moving closer to each other.
TG_NEAR_PARREL_MOVE_UP	Two fingers are moving together upwards.
TG_NEAR_PARREL_MOVE_DOWN	Two fingers are moving together downwards.
TG_NEAR_PARREL_MOVE_RIGHT	Two fingers are moving together to the right.
TG_NEAR_PARREL_MOVE_LEFT	Two fingers are moving together to the left.

Some of the interpreted gestures are further analyzed as described:

- **TG CLICK:** One of the most used events; once a click is performed, the system will check the position of where it was performed. An analysis is done to determine if the click was made on a button of the GUI. If not, the current state of the system is determined in order for the controller to know how to interpret the event (either of the tool panel was touched, a tool annotation was selected or placed, etc.).
- **TG MOVE RIGHT:** The system current state is verified. If the system is in line drawing mode, the point is reported as part of the currently drawn line. If not, the point is considered as part of the region drawn in order to select lines later on. This is also true for the TG_MOVE_LEFT, TG_MOVE_UP and TG_MOVE_DOWN events, but in their respective directions.
- **TG TOUCH END:** All the events ended. Depending of previously done tasks, the system gives the signal to start the process of saving the drawn line, sending the new position of the annotations or start the line selection process.
- **TG ROTATE CLOCK:** Rotate clockwise the selected annotations (either tool or line). This is also done counter clockwise by TG_ANTIROTATE_CLOCK.
- **TG NEAR PARREL MOVE UP:** Translate the selected annotations upwards. The other events handle the other directions.

As a part of the installation process, all the previous work done regarding the line annotation creation needed to be adapted: instead of creating the lines using a keyboard event, as it previously was, the lines needed to be created and drawn on the screen as the user made the required touch event. The result of the adaptation is presented in Figure 5.



Figure 5. Line annotation being drawn with a touch event

System Refining and Correction

Once the touch overlay was installed and working, work was done on fixing some of the bugs that inhabited the code before continuing the development of new features. Those tasks included:

- **Rendering flickering effect**: Because of the way in which the rendering of the image acquired through the network was done, the line annotations experienced a flickering effect when they were drawn in the screen. This was addressed by creating a special case before the scene redrawing process, so that the color of certain pixels was not modified during the process.
- **TCP-IP communication refining**: All the code that was used for the TCP-IP communication was not as modular as desired. Because of that, corrections were made that allowed the Mentor System to have several sockets opened, either for data reception or dispatch.
- **General workflow controller**: In order to preserve the modularity of the system, a way to control the whole workflow of it needed to be created. Because of that, a general controller was programmed: it consists of a final state machine with several flags that represent the state of the system. All the modules of the system check the state of the flags before performing their routines, and refresh them when done.
- **Coordinate system corrections**: In preparation for the integration with the tablet based Trainee System, the way in which the coordinate system was structured was modified. A more safe and modular approach was taken, in which the resolution of the system was normalized so that the annotations and transformation could be replicated, no matter the resolution of the target Trainee System.
- **Point annotations creation**: The functionality of creating circle-shaped line annotations representing points of interest was added. If the point line creation mode is enabled and a click event occurred, a circle will be drawn using the event location as the center of the annotation.
- **Line structure redefinition**: Before the touch overlay was installed, all the lines were treated as a single element when a geometrical transformation was applied, all the transformations were done using the center of the image as an anchor point and there was no way of selecting lines independently. This approached proved to be insufficient to the tasks that the Mentor System was supposed to perform. Because of that, the whole approach was changed and a specific object to represent each line was created. The specific position of each line can now be determined (as the center of the line is calculated each time a transformation is applied) and they can be selected, edited and erased independently. This is demonstrated in **Error! Reference source not found.** and Figure 7.

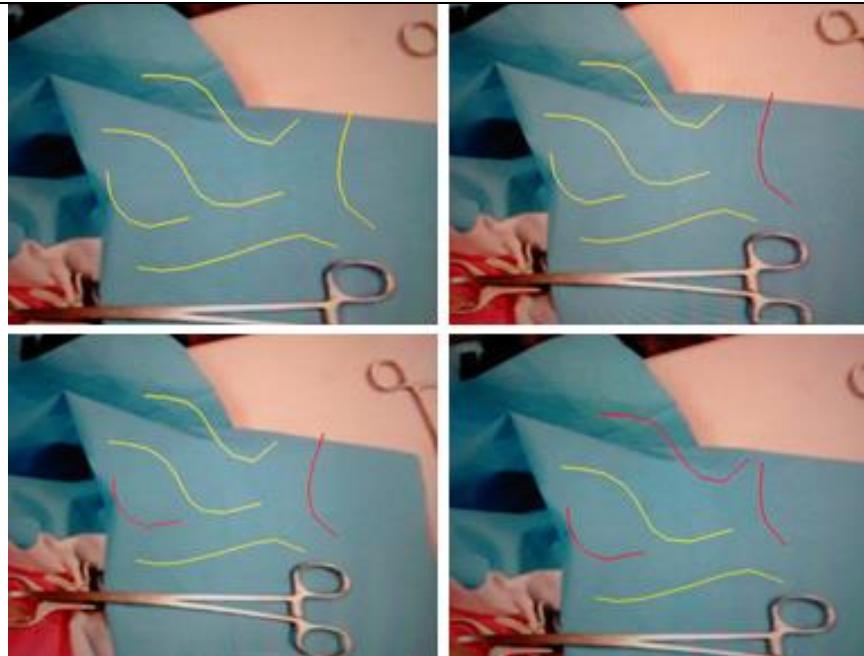


Figure 6. Group of line annotations selection

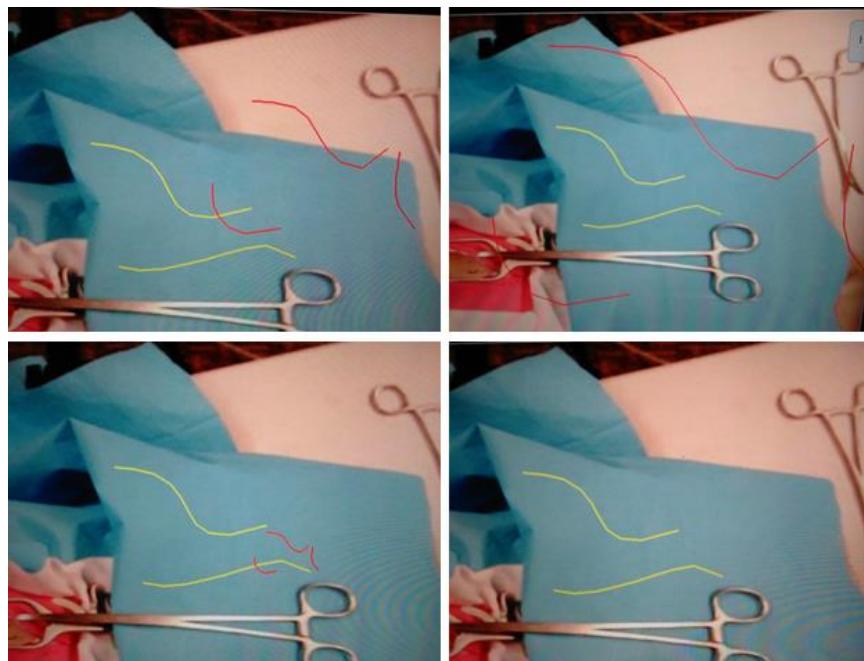


Figure 7. Editing and erasing groups of line annotations

Graphic User Interface Creation

Since the system was supposed to be not just useful but usable, a GUI for it needed to be created. A research about how to integrate an OpenGL context window within a window using a pre-built user interface library was done. Several C++ options were found (such as SFGUI, MyGUI, nanogui and a Windows Forms approach). After some testing, all of them implied a

long adaptation process for the already created OpenGL windows to be integrated within their context. Another approach was taken after this: instead of doing a migration of the OpenGL context, a GUI pre-created image will be overlapped with the image received from the network before initiating the rendering process. Buttons and a panel containing the tool annotations were created, which are presented in Figure 8.

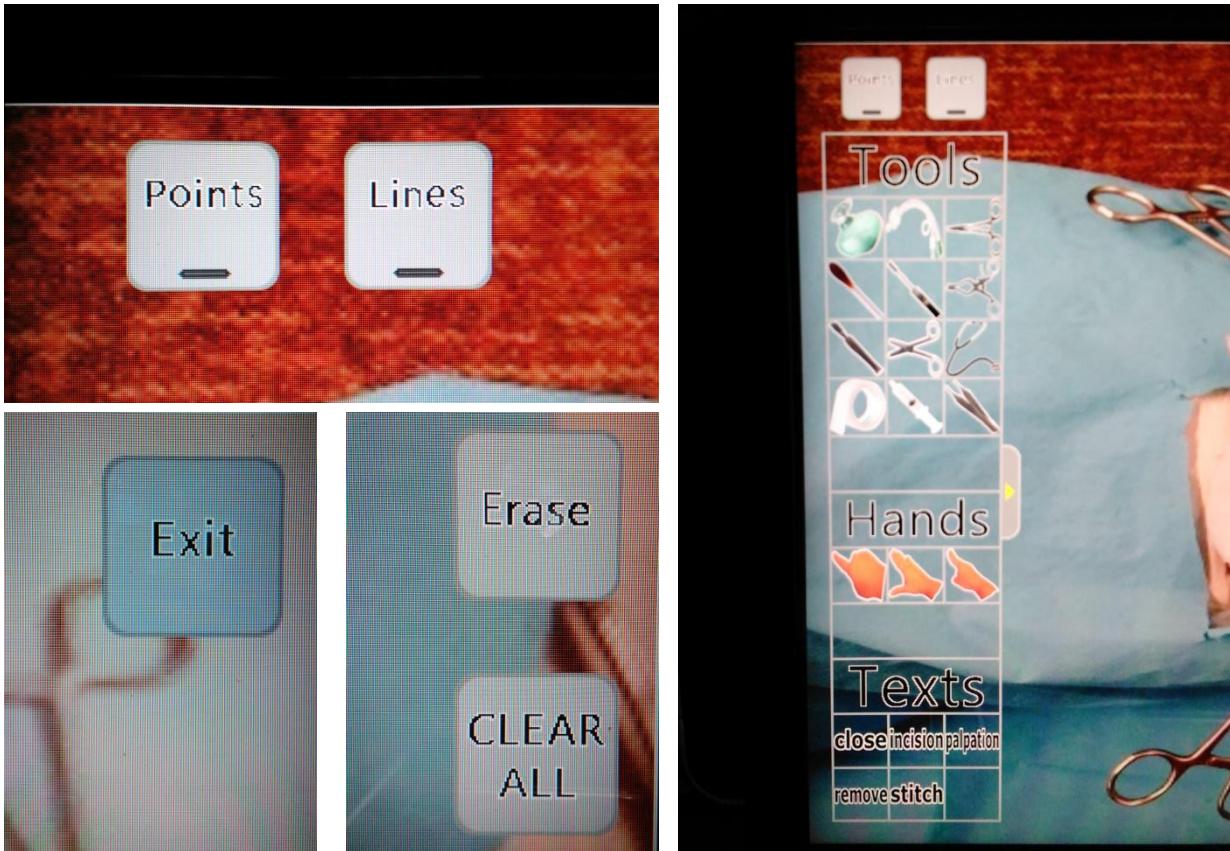


Figure 8. Mentor System Graphic User Interface

Because the GUI is just created by overlaying images on top of another one, it does not have any real buttons to be pressed. To emulate this, extra processes were created on the click touch event recognition: as soon as the overlay receives a touch event, it analyzes if the event was done over the coordinates of one of the buttons. If that is the case, it activates one of the flags of the workflow controller so that the other modules can perform the operations that touching that button implied. An example of how the GUI can change by doing a click can be found in Figure 9.

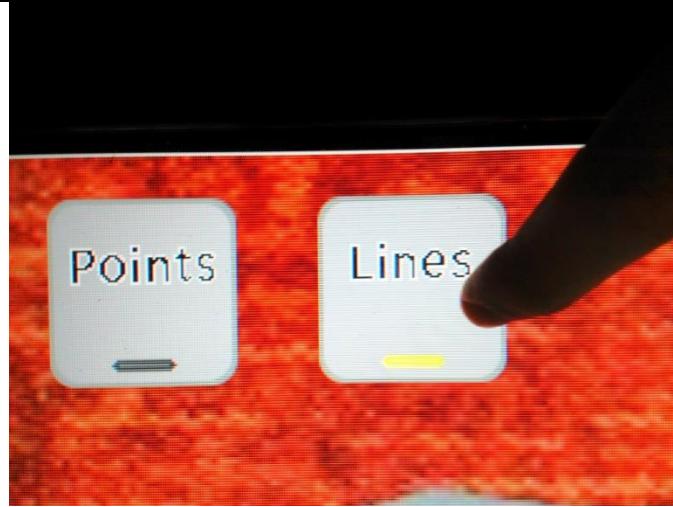


Figure 9. Enabling the line annotation drawing mode by clicking a GUI button

Virtual Tool Annotations Creation

Another of the features that the Mentor System was supposed to have was the ability of creating, selecting, editing and erasing virtual tool annotations. Forty images representing diverse surgical instruments, hands positions used by surgeons and words symbolizing actions performed during surgery were created. The main goal was to successfully overlay those annotations over the image received from the remote surgical room.

The process is divided in various steps: as soon as the system first receives a click inside of the tool annotation panel, it detects which one of virtual annotations was clicked. Once the specific tool annotation gets determined, a system starts waiting for another click to be made in the screen: the next place clicked (unless it is a button) will become the anchor point of the image (each image has an anchor point that symbolizes the place in which a real tool will make contact with a patient). The tool annotation is defined and its internal values (zoom, rotation, the .PNG corresponding to that specific annotation) are initialized.

Before the rendering process starts, the system continuously goes through all the created tool annotations (if any) and retrieves their anchor points location (which gets edited by translating the annotation) and their images (when the image is retrieved, the zoom and rotation processes are applied). Finally, the retrieved image gets overlaid on top of the generated GUI image. The annotations can be selected by clicking a pixel that is part of the sprite of the annotation, allowing the system to have multiple annotations at the same time and manipulate them independently.

Internally, the virtual space in which the image is drawn is bigger than the actual size of the image. This is done so that when rotating the image around its anchor point, the created does not get cropped.

Some examples of the usage of the virtual tool annotations are shown in Figure 10.

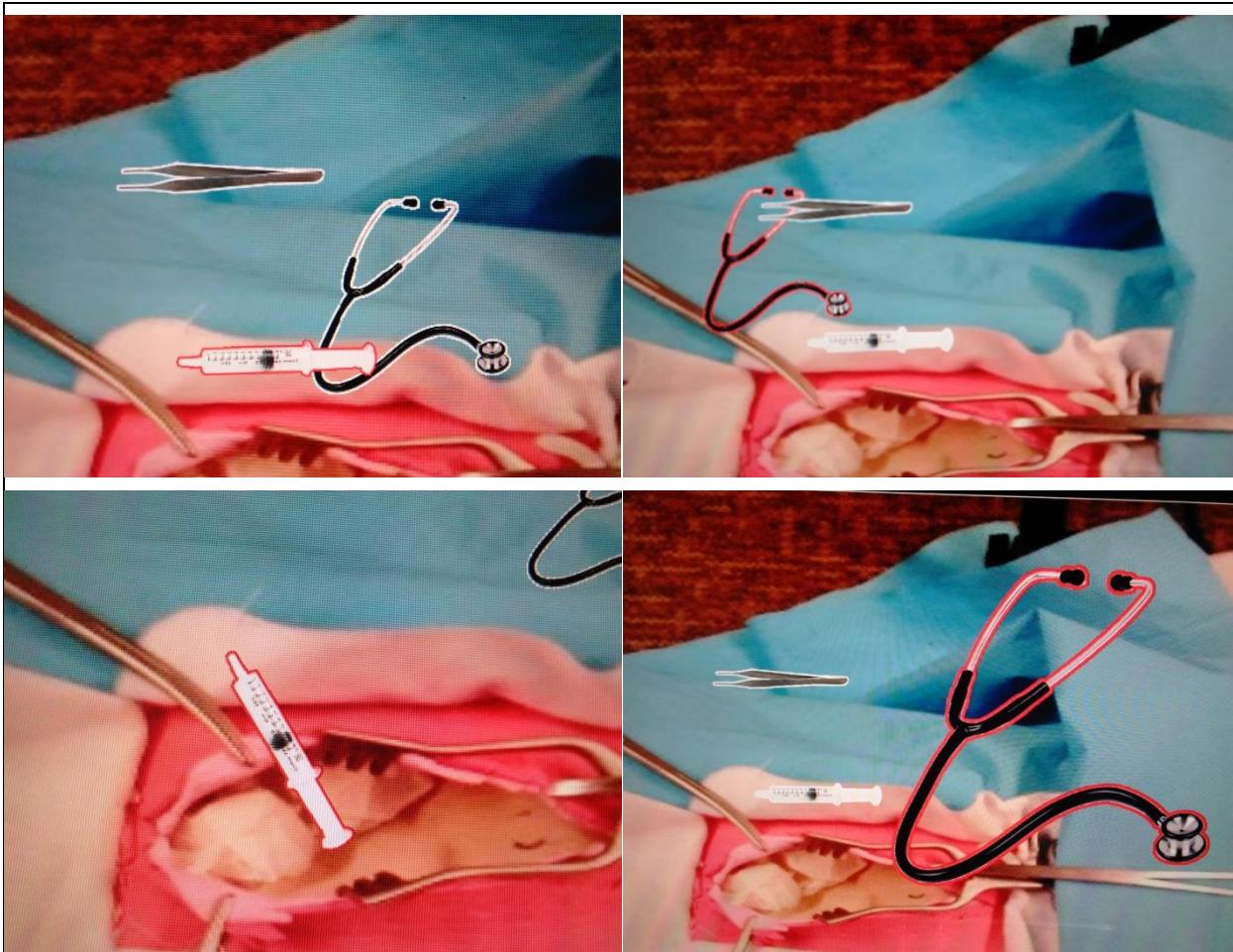


Figure 10. Creation and edition of virtual tool annotations

JSON Communication Protocol

The Mentor System had to be able to communicate through the network with the Trainee System, which could be hundreds of miles away. Because of the large variety of event and signal that the Mentor System is able to create, a stable protocol to successfully communicate the Mentor System with the tablet-based Trainee System needed to be created. Thanks to its modularity, the JavaScript Object Notation (JSON) was the option selected to build the whole communication protocol around off. JSON, originally made as an aid for JavaScript during browser/server communication, is a lightweight data-interchange format used to exchange information using human-readable text by sending them inside of attribute-value pairs. The JSON will be created with a specific format, depending on what it represents: a line annotation, a tool annotation or a delete event. Both the Mentor and Trainee Systems work with the same protocol, ensuring that both ends will know how to interpret the data that they are sending/receiving.

Error! Reference source not found. shows a generated JSON file of the creation of a hemostat tool annotation, which uses the general format for a virtual tool annotation:

```
{
  "id":2,
  "command":"CreateAnnotationCommand",
  "annotation_memory":{
    "matches":{},
    "initialKeyPoints":[],
    "initialDescriptors":{},
    "initialAnnotation":{
      "annotationPoints":[
        {
          "y":85.25390625,
          "x":100.15625
        }
      ],
      "scale":0.30000001192092896,
      "annotationType":"tool",
      "toolType":"hemostat",
      "rotation":0,
      "selectableColor":-12580371
    },
    "currentAnnotation":{
      "annotationPoints":[
        {
          "y":85.25390625,
          "x":100.15625
        }
      ],
      "scale":0.30000001192092896,
      "annotationType":"tool",
      "toolType":"hemostat",
      "rotation":0,
      "selectableColor":-12580371
    },
    "currentHomography":{},
    "initialRawKeyPoints":[],
    "currentRawKeyPoints":[]
  }
}
```

Figure 11. Creation and edition of virtual tool annotations

After the JSON files were successfully written and read, both systems created a channel for them to send/receive the JSON files between each other. The communication module on the Mentor System side went through some changes in order for it to be able to listen to multiple connection sockets at all the time.

Air Gestures Communication Channel

One of the main goals of the Mentor System is to be able to interpret air gestures performed by surgeons and to do the necessary routines after interpreting them. Because of the changes done to the communication scheme of the Mentor System, it now can easily have several channels opened and listening at the same time for different type of inputs. In an attempt of keeping the system as modular as possible, the approach taken leverages the changes done over the communication module.

Another communication channel was opened to get information from another client system running on the computer executing the gesture recognition algorithms. Once a gesture prediction occurs, a string is sent through the communication channel to the Mentor system which will be listening for those commands. Currently, the received messages are only being displayed on the Mentor display. This is work in progress, and as functionality gets developed for the interaction display, actions conveyed by gestures will be calling said functions to execute commands on the Mentor System.

System Documentation

One of the most important parts on any project is to document all the progress and code done. The Mentor System has all of its functionalities and routines documented in-code, and a hard documentation is also being developed. The goal of all the documentation of the system is for the system to be operated by anyone, even if that person just has basic programming skills. Diagrams that further explain the system are also being developed. With these diagrams, the general workflow of the Mentor System can be easily understood and the place to perform corrections and optimizations can be easily spotted through the code.

As an example, the basic structure that describes how the Mentor System modules were built altogether is presented in Figure 12.

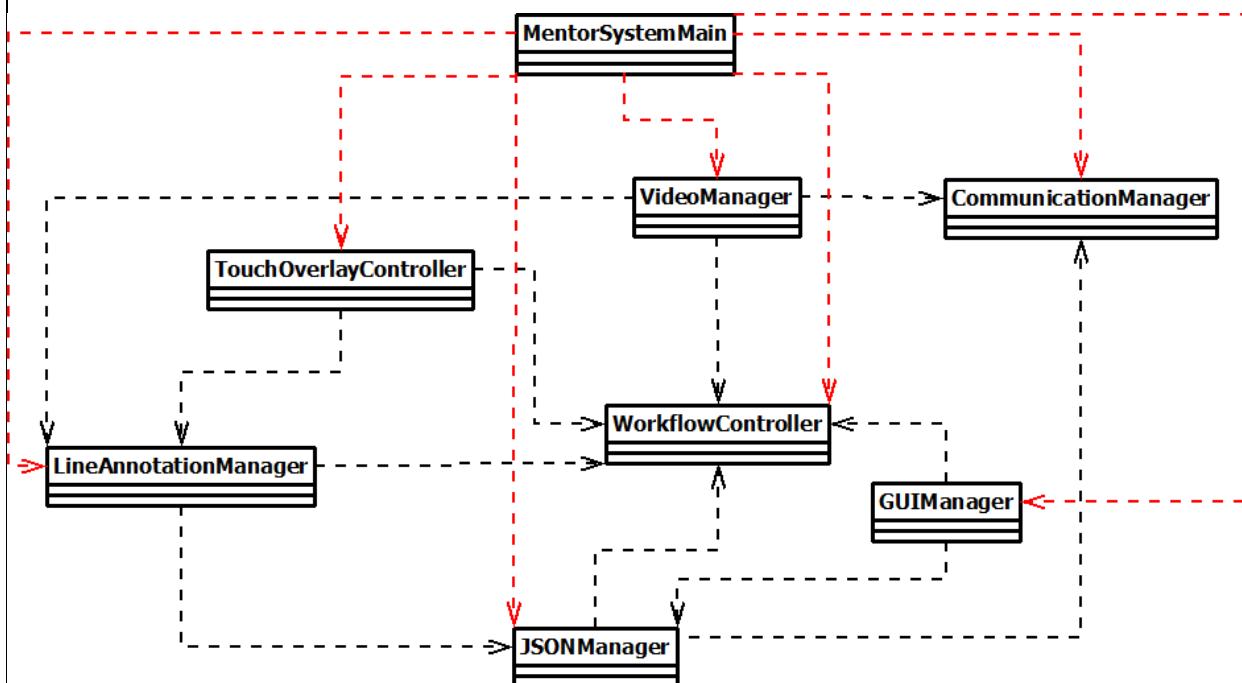


Figure 12. Mentor System general code structure

Mentoring channel between trainee surgeons and expert surgeons

After putting all the required modules and performing extensive tests over them, the Mentor System was ready to be joined with the tablet-based Trainee System. After some meetings to put everything altogether, the connection between the systems worked alright. Figure 13 illustrates both systems working together, communicating with each other via Wi-Fi:

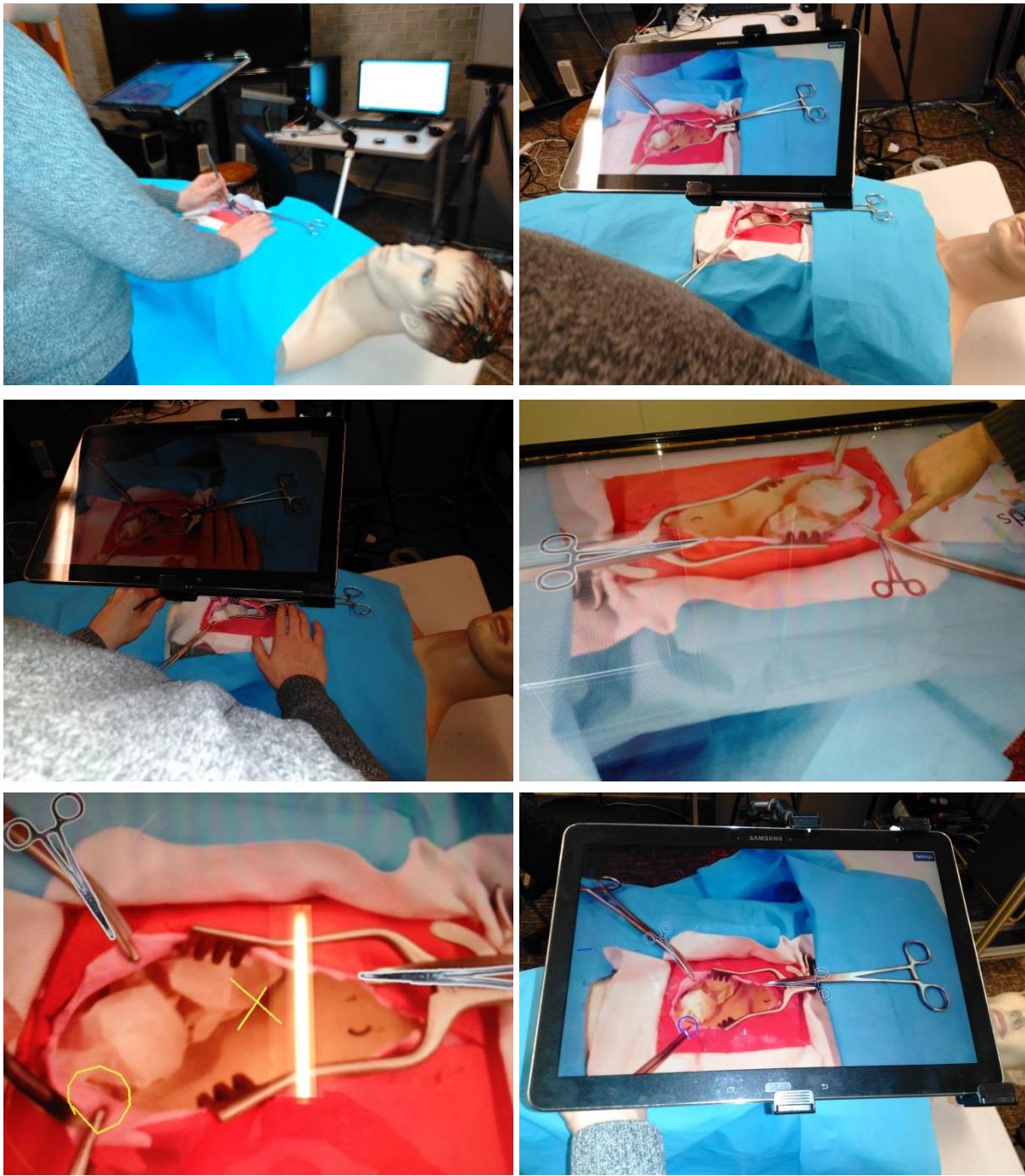


Figure 13. Patient-size Mentor System and Tablet-based Trainee System working together through Wi-Fi

Figure 14 demonstrates how a person using the Trainee System can replicate the instructions sent by the Mentor System:

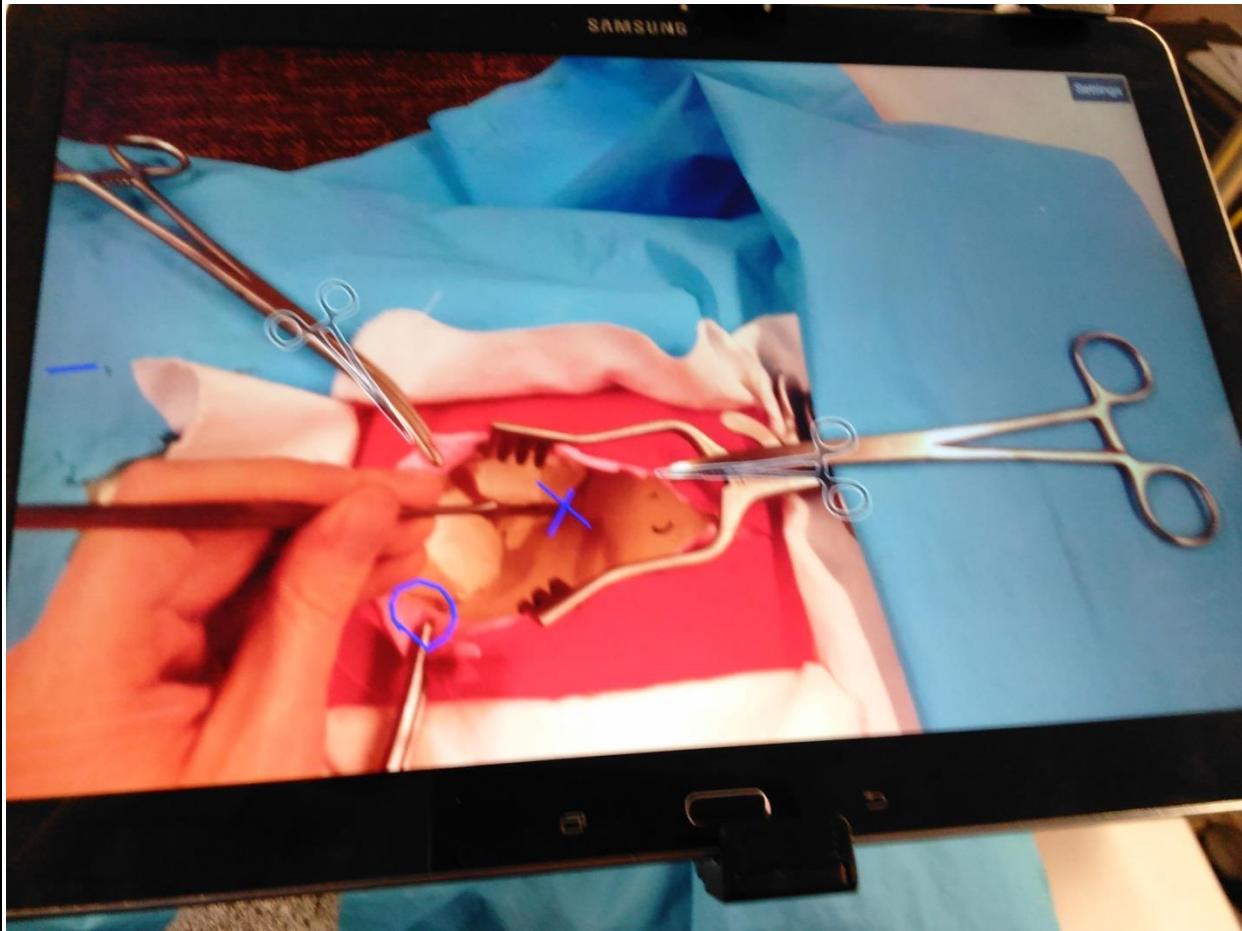


Figure 14. Replication of mentor instructions at the Trainee System side

The following link to a video illustrates the functionalities that can be made by using the Mentor System: <https://youtu.be/Y28Zo0f8oI8>

International Meeting on Simulation in Healthcare (IMSH) 2016

As an effort to advertise the project and to show the advancements done on it, the project team attended the IMSH 2016 at San Diego, CA. During this meeting, the system was part of the DOD funded-projects corral, in which different projects showcased their progresses to the public attending the conference. The STAR Project was widely accepted by the community attending the conference and several contacts were gathered, confirming the importance and potential that this system can have in the medical community.

What opportunities for training and professional development has the project provided?

From January 16 to 20, several members of our team attended the IMSH 2016 conference in San Diego in order to demonstrate our current STAR prototype system as part of the Department of Defense's Research Corral. We used a printed anatomical poster, placed on a table, as a patient simulator, and set up the STAR trainee tablet system over the poster. Imagery was sent over to the mentor tablet system, which was also present. In this way, visitors to the booth were able to see the user interfaces of both mentor and mentee systems. Visitors were also invited to mark indicated locations on the poster under telementored instruction, demonstrating through an interactive process how the STAR approach to reducing focus shifts can improve telementoring. We were able to speak with a large number of attendees from a wide array of backgrounds (surgeons, nurses, software engineers, military officers, etc). As a result, we were able to gain new insight into potential use cases of our system, and we were able to disseminate our research to a larger audience.

How were the results disseminated to communities of interest?

On October 16, 2015, we presented at the Eskenazi Trauma Symposium in Indianapolis, IN. The presentation was entitled "STAR: Using Augmented Reality Transparent Displays for Surgical Telementoring." The audience was largely resident medical students, nurses, physicians, and surgeons, few of whom had existing expertise in augmented reality or computer graphics research. This presentation described the importance and potential of telementoring, described STAR's approach toward telementoring, and described the system's validation in user studies.

On October 20, 2015, we gave a presentation to the Purdue graphics lab's "Graphics Lunch," which is a gathering of faculty, lab members, and students interested in computer graphics

research. The presentation was entitled “Creating a Magic-Lens Transparent Display Effect on a Tablet” and described the team’s work on simulated transparent displays and their relation to surgical telementoring.

We gave another presentation to the Graphics Lunch on February 3, 2016, entitled “Simulated Transparent Displays: Implementation and Analysis.” This presentation focused on the team’s efforts to define measurements of transparency error, and on the use of theoretical and empirical error bounds to determine the priorities for future research into improving a transparent display effect.

In January 2016, a journal paper entitled “A Hand-Held, Self-Contained Simulated Transparent Display” was submitted to the SIGGRAPH 2016 conference. This paper describes the simulated transparent prototypes we have created as well as our analysis of transparent display error. However, after receiving initial reviews we determined that SIGGRAPH would not be a good fit for the work we have done. After withdrawing the submission to SIGGRAPH, we resubmitted a version of this paper to the ISMAR 2016 conference.

What do you plan to do during the next reporting period to accomplish the goals?

Task 1.1 – Implement transparent display

We will use the knowledge we have gained from our research into simulated transparency to improve our telementoring system’s model of the operating field. Currently, the operating field is treated as a 2D image and all annotation anchoring is done by screen-space transformations. We will integrate 3D geometric models of the operating field so that acquired imagery is registered in a common world-space coordinate system. At first, this model will still be 2D and planar, but the changed representation will allow the mentor system to be able to move a virtual camera in relation to the mentee’s acquired imagery of the operating field. For example, the mentor user will be able to zoom in/out and pan the operating field imagery, rather than just zooming/panning the annotations. We will then investigate modeling 3D meshes to overlay onto the operating field, which will allow virtual annotations to interact with the mesh in complex ways.

Task 1.2 – Achieve visual overlay of information

We will implement the proposed architecture of future simulation, where a series of real-world video clips of surgical steps can be overlaid and parameterized onto the mentee’s current view of the operating field. We will acquire a small test set of videos that we can process and integrate into the system. The result will be a first approximation of future simulation in a 2D planar visualization. Next, we will investigate simulation of future surgical steps in the context

of 3D geometry. By researching how deformation of the operating field can be detected and modeled, we plan to be able to have 3D animated annotations that show the mentee how certain incisions will deform and change the structure of underlying tissue.

Task 2.1- Develop a gesture-based interaction system

Subtask 2.1.1 – One-Shot Learning Gesture Recognition

The next objectives on this task are related with expanding the feature representation of the gesture to include hand poses. In this sense, more gestures found during our data collection experiment will be able to be recognized and implemented. Additionally, future work will include incorporating conveying actions and performing routines on the Mentor Interaction System, based on the recognized gestures sent through the available communication channel.

Subtask 2.1.2 – Design a projection surface and interaction methodology

Now that the prototype patient-size touch surface is completed, further experiments using it need to be conducted. A test case that demonstrates the capabilities of the system will be developed and performed by several test subjects, so that a large base of usage experiences will be created. Once this data is analyzed and the necessary corrections over the system are done (if any), more complex texts involving the medical branch of the project (e.g. real surgeons) will be performed.

4. IMPACT:

What was the impact on the development of the principal discipline(s) of the project?

This technology will increase the sense of co-presence in the operating room between mentor and trainee. This is a fundamental step towards telexistence. Telexistence is a concept used to describe the framework that allows humans to have a real-time sensation of being and interacting with objects in places somewhere different from their actual location. The fundamental premise is that a higher sense of co-presence has an impact on the quality of mentorship. For example, by allowing the mentors to physically interact with the patient's anatomy through hand gestures (embodied interaction), the mentor's level of immersion and engagement will be significantly increased.

What was the impact on other disciplines?

In this period we completed the second experimental design, which consists of collecting the gestures that mentors perform while interacting with the large projection table. It is expected that the use of gestural interfaces and the gesture lexicon design will increase the understanding about the different uses of nonverbal communication in the operating room, with extensions to other high-risk/ high-stakes scenarios.

What was the impact on technology transfer?

We requested a temporal patent based on the concepts described on this report.

What was the impact on society beyond science and technology?

Currently the main instrument to improve surgical skills in trauma surgery requires animal models, one to one mentorship and lengthy and complex training sessions (e.g. the ATOM course attended by the PIs of this project). A more cost effective option that will make this training scalable consists of having the training surgeon teach the same ATOM class, remotely, through the STAR platform. This will allow tens residents (current there are only 10-15 per class) to participate concurrently with only one mentor

5. CHANGES/PROBLEMS: The Project Director/Principal Investigator (PD/PI) is reminded that the recipient organization is required to obtain prior written approval from the awarding agency Grants Officer whenever there are significant changes in the project or its direction. If not previously reported in writing, provide the following additional information or state, "Nothing to Report," if applicable:

Changes in approach and reasons for change

There were no significant changes in our approach during this period. One minor change was that we changed our mentor system to use a new codebase for a Windows machine rather than our existing Android mentor tablet system. We did this in order to be able to use the multi-touch system of the interaction table, which was unable to interface properly with the Android tablet system.

Actual or anticipated problems or delays and actions or plans to resolve them

One minor delay was a result of our work in re-implementing the mentor system in a new codebase on a Windows machine (to replace our existing tablet-based Android mentor system). Some inconsistencies in the network protocols being used required some additional time for our software developers to determine a consensus for the protocols. After a few days of system architecture planning, development was able to continue without much delay. There is also some delay in the completion of the simulated of the future steps in surgery. Work has started in this task, but have not yet been completed.

An anticipated problem on the gesture recognition system implemented, is the current limitation to work with gross gestures, which is why future work includes incorporating features to represent hand poses. The use of a different sensor to gather that type of data may be considered.

Changes that had a significant impact on expenditures

No changes

Significant changes in use or care of human subjects, vertebrate animals, biohazards, and/or select agents

Significant changes in use or care of human subjects

No changes

Significant changes in use or care of vertebrate animals.

No changes

Significant changes in use of biohazards and/or select agents

No changes

• **Publications, conference papers, and presentations**

Report only the major publication(s) resulting from the work under this award.

Journal publications.

(Andersen et al., 2016)

Andersen, D., Popescu, V., Cabrera, M. E., Shanghavi, A., Gomez, G., Marley, S., ...

Wachs, J. P. (2016). Medical telementoring using an augmented reality

transparent display. *Surgery, In*

Press(<http://dx.doi.org/10.1016/j.surg.2015.12.016>).

<http://doi.org/10.1016/j.surg.2015.12.016>

Daniel Andersen, Voicu Popescu, Maria Eugenia Cabrerea, Aditya Shanghavi, Gerardo Gomez, Sherri Marley, Brian Mullis, Juan Wachs. "An Augmented Reality Based Approach for Surgical Telementoring in Austere Environments." *Journal of Military Medicine*. 2015 (submitted). Acknowledgment of federal support: yes.

Daniel Andersen, Voicu Popescu, Chengyuan Lin, Maria Eugenia Cabrerea, Aditya Shanghavi, Juan Wachs. "A Hand-Held, Self-Contained Simulated Transparent Display." *ISMAR* 2016 (submitted). Acknowledgment of federal support: yes.

Books or other non-periodical, one-time publications.

Other publications, conference papers, and presentations

Daniel Andersen. "STAR: Using Augmented Reality Transparent Displays for Surgical Telementoring." Eskenazi Health 22nd Annual Trauma & Surgical Critical Care Symposium. Indianapolis, IN. 16 Oct 2015. Conference Presentation.

Daniel Andersen, Voicu Popescu, Maria Eugenia Cabrera, Aditya Shanghavi, Edgar J. Rojas Muñoz, Brian Mullis, Sherri Marley, Gerardo Gomez, Juan P. Wachs. "STAR - A System for Telementoring with Augmented Reality." Demo Exhibit in Government Agency R&D Corral at IMSH 2016.

- **Website(s) or other Internet site(s)**

<https://engineering.purdue.edu/starproj/> - Official project website, with overview of research, links to publications, images, and videos.

- **Technologies or techniques**

The technique for one shot gesture recognition is a result from the research activity. It is based on the idea that gestures have a simplified compact representation that can be easily stored and on the fact that human-made gestures are constrained by the bio-mechanical and anthropometric features of the human body. With such representation and contextual knowledge, meaningful variability can be incorporated while generating a larger example dataset which can later be used to train traditional classifiers.

- **Inventions, patent applications, and/or licenses**

We filled a temporal patent with the prototype of the STAR system that we developed.

-

Databases, videos, raw images and recording of the ATOM sessions (3) are located at the PURR repository.

<https://purr.purdue.edu/projects/starproject/files/>

7. PARTICIPANTS & OTHER COLLABORATING ORGANIZATIONS

What individuals have worked on the project?

Example:

Name: **Mary Smith**
Project Role: **Graduate Student**
Researcher Identifier (e.g. ORCID ID): **1234567**
Nearest person month worked: **5**

Contribution to Project: **Ms. Smith has performed work in the area of combined error-control and constrained coding.**

Funding Support: **The Ford Foundation (Complete only if the funding support is provided from other than this award).**

<i>Name:</i> Juan P Wachs <i>Project Role:</i> Principal Investigator <i>Researcher Identifier (e.g. ORCID ID):</i> 0000-0002-6425-5745 <i>Nearest person month worked:</i> 1.12 month	<i>Contribution to Project:</i> Supervising the overall performance of the project. Coordinated visits to IUSM. Working with Maria Eugenia in all the aspects of gesture recognition and one shot learning. Working with Aditya Shanghavi for the design of the large interaction table. Helping with the journal publication.
<i>Name:</i> Voicu Popescu <i>Project Role:</i> Co-Investigator <i>Researcher Identifier (e.g. ORCID ID):</i> 0000-0002-6425-5745 <i>Nearest person month worked:</i> 1.12 month	<i>Contribution to Project:</i> Actively participated in and advised research assistant Daniel Andersen in the research and development of the first prototype of the augmented reality transparent display surgical telementoring system (i.e. the STAR platform); in designing, conducting, and analyzing the results of user studies aimed at assessing STAR; in disseminating the project results in a journal paper.
<i>Name:</i> Gerry Gomez <i>Project Role:</i> Co-Investigator <i>Researcher Identifier (e.g. ORCID ID):</i> 0000-0002-6425-5745	

<i>Nearest person month worked:</i>	2 weeks
<i>Contribution to Project:</i>	<i>Provided formative feedback about the first and second prototype. Conducted the ATOM course and described throughout the course the context of our system. Acted as the mentor in the initial test at IUSM and provided knowledge about the cric procedure.</i>
<i>Name:</i>	<i>Brian Mullis</i>
<i>Project Role:</i>	<i>Co-Investigator</i>
<i>Researcher Identifier (e.g. ORCID ID):</i>	
<i>Nearest person month worked:</i>	
<i>Contribution to Project:</i>	<i>Provided formative feedback about the applicability of the prototype to austere environments, and specifically its benefits and drawbacks when used for orthopedic surgery. He also provide assistance regarding the fasciotomy procedure and the possibility to show case this procedure in Experiment 2, in a simulated environment.</i>
<i>Name:</i>	<i>Sherry Marley</i>
<i>Project Role:</i>	<i>Co-Investigator</i>
<i>Researcher Identifier (e.g. ORCID ID):</i>	
<i>Nearest person month worked:</i>	
<i>Contribution to Project:</i>	<i>Helped the Purdue team with the experimental design. Coordinated the attendance to the ATOM course three times. She provided consultancy regarding the surgical training process and actionable knowledge during the cric.</i>
<i>Name:</i>	<i>Dan Andersen</i>
<i>Project Role:</i>	<i>Research Assistant</i>
<i>Researcher Identifier (e.g. ORCID ID):</i>	
<i>Nearest person month worked:</i>	<i>5.25 months</i>
<i>Contribution to Project:</i>	<i>Responsible for architecting, programming and developing tablet system for mentor and trainee tablets. Researched and implemented feature detection / descriptor matching approach for current annotation anchoring algorithm. Was major contributor to journal</i>

<p><i>paper (currently under review) demonstrating the STAR system. Contributed to planning and conducting ongoing user studies to validate system.</i></p>	
<i>Name:</i>	<i>Maria Eugenia Cabrera</i>
<i>Project Role:</i>	<i>Research Assistant</i>
<i>Researcher Identifier (e.g. ORCID ID):</i>	
<i>Nearest person month worked:</i>	<i>5.25 months</i>
<i>Contribution to Project:</i>	<i>Maria Eugenia worked together with Dan in the experimental design, recruitment of human subjects, development of the testing environment and mock surgical scenarios. She is now working on the one-shot learning concept for gesture recognition.</i>
<i>Name:</i>	<i>Aditya Ajay Shanghavi</i>
<i>Project Role:</i>	<i>Master Student</i>
<i>Researcher Identifier (e.g. ORCID ID):</i>	
<i>Nearest person month worked:</i>	<i>3 months</i>
<i>Contribution to Project:</i>	<i>Aditya designed the projection table, tested different projection materials, and types of projectors in order to project a whole silhouette in the table. Aditya also implemented the Gooseneck and the tablet holder and the adaptor to the WAM robotic arm.</i>
<i>Name:</i>	<i>Edgar Rojas</i>
<i>Project Role:</i>	<i>Under grad Student</i>
<i>Researcher Identifier (e.g. ORCID ID):</i>	
<i>Nearest person month worked:</i>	<i>3 months</i>
<i>Contribution to Project:</i>	<i>Edgar developed the mentoring system architecture together with the software and libraries required to interact with the large display</i>

Has there been a change in the active other support of the PD/PI(s) or senior/key personnel since the last reporting period?

Juan Wachs	09/01/2014 - 08/31/2017	0.23 SU 0.5 AY
<p>University Of Denver NSF: MRI Development: Human Avatars: Enabling Research in Natural Communication with Virtual Tutors, Therapists, and Robotic Companions</p>		
<p>Major Goals of the Project: The goal of the proposed MRI development project is to develop a life-like emotive software/hardware instrument in the form of robotic character heads that will support natural spoken dialogs between the robot and a human that closely models the face-to-face communication behaviors of a sensitive and effective human tutor, clinician or caregiver to a degree unachievable with current instrumentation.</p>		
<p>Overlap: No overlap.</p>		
Juan Wachs	09/5/2014 - 08/31/2019	0 SU 0 AY
<p>NSF: Collaborative Research: I/UCRC for Robots and Sensors for the Human Wellbeing</p>		
<p>Major Goals of the Project: The goal of the proposed center is to develop technology in the form of robots and sensors for assistive technologies to support therapies and rehabilitation of people with disabilities.</p>		
<p>Overlap: No overlap.</p>		
Juan Wachs	04/1/2015 - 03/31/2016	0.12 SU 0.5 AY
<p>THE NAVSUP FLEET LOGISTICS CENTER SAN DIEGO: An Efficient Real-Time Method for Detection and Characterization of UAVs</p>		
<p>Major Goals of the Project: The research objective of this proposal is to develop a video-based methods for real-time detection of small, unmanned aerial vehicles (UAVs) leveraging on effective sense and avoid techniques. Such methods can be integrated into real-time on board</p>		

processors. This, in turn, would lead to enhanced UAV's capabilities for detection of friendly and unfriendly airborne traffic and respond with appropriate alarms, maneuvers and notifications.

Overlap: No overlap.

What other organizations were involved as partners?

Organization Name: Indiana University School of Medicine

Location of Organization: Indianapolis, USA

Partner's contribution to the project (identify one or more)

- *Experimental Design for experiment 2. The co-Investigators helped on the design of the fasciotomy experiment, provided the supplies and supported the completion of the experiment.*
- *In-kind support: they made available the surgical instruments and facilities to complete Experiment 2*
- *Collaboration: Dr. Gomez, Mrs. Marley and B. Mullis collaborated with the project staff on the project);*
- *Personnel exchanges: We visited IUSM for Experiment 2 and the grad students participated in the discussions and experiments.*

8. SPECIAL REPORTING REQUIREMENTS

COLLABORATIVE AWARDS:

QUAD CHARTS: N/A

9. APPENDICES: N/a